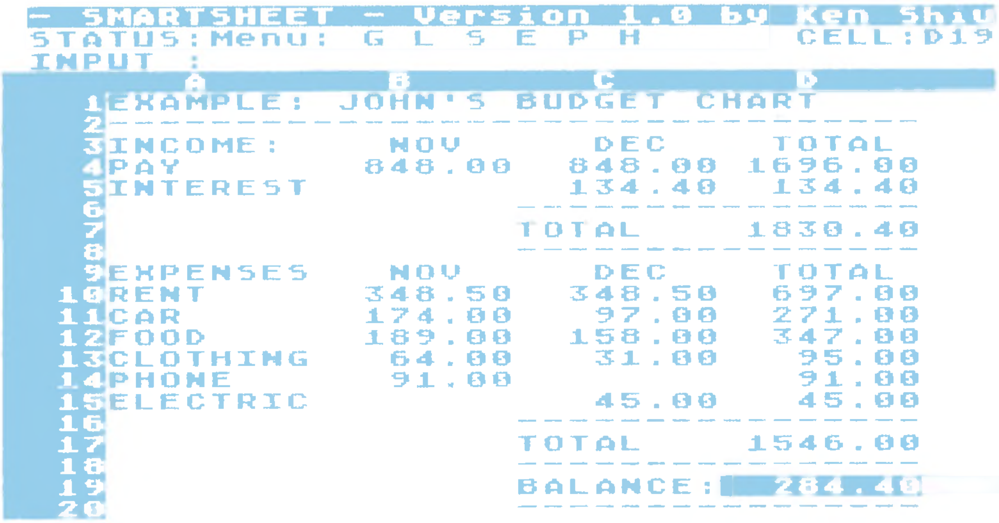
ATARI COMPUTER ENTHUSIASTS (N.S.W.)

A.C.E. (N.S.W.)
G.P.O. BOX 4514,
SYDNEY. 2001.
N.S.W. AUSTRALIA.

INSIDE INFO No. 19 June 1985



```
 - SMARTSHEET - Version 1.0 by Ken Shiu
 STATUS:Menu: G L S E P H      CELL:D19
 INPUT  :
        A       B       C       D
   1EXAMPLE: JOHN'S BUDGET CHART
   2--------------------------------
   3INCOME:    NOV     DEC    TOTAL
   4PAY       848.00  848.00 1696.00
   5INTEREST          134.40  134.40
   6                ----------------
   7                TOTAL    1830.40
   8                ----------------
   9EXPENSES   NOV     DEC    TOTAL
  10RENT      348.50  348.50  697.00
  11CAR       174.00   97.00  271.00
  12FOOD      189.00  158.00  347.00
  13CLOTHING   64.00   31.00   95.00
  14PHONE      91.00           91.00
  15ELECTRIC           45.00   45.00
  16                ----------------
  17                TOTAL    1546.00
  18                ----------------
  19                BALANCE:  284.40
  20                ----------------
```

- SOFTWARE EXCHANGE NEWS
- ATARI 1020 PRINTER HANDLER
- ASSEMBLY LANGUAGE
- ANTENNA ANALYSIS
- TOUCH TABLET APPLICATIONS
- BOOK REVIEW
- DOCTOR DOS
- FILE TRACER
- ADVENTURE!
- SMARTSHEET
- LOGO
- LABEL PRINTER
- FORTH
- SOUND EFFECTS
- DOLLAR FORMATTING

PRICE $3.00

REGISTERED BY AUSTRALIA POST PUBLICATION No. NHB 6844

EDITORIAL

I thought it would be a nice touch to have the June INSIDE INFO slightly larger and here it is! In INSIDE INFO No.17, I had hoped for the theme of the June issue to be Home Finance and Applications. Instead, this issue has seen the birth of some new columns. I have been giving serious thought to the format of INSIDE INFO. Perhaps there should be more regular columns catering for special interests, e.g. Adventure!, Assembly Language Programming. Should there be an arcade game every issue? I would like to know how members feel about INSIDE INFO and how it could be modified to best suit everyone's needs. I need FEEDBACK!!!

Congratulations must go to Peter Bamford for the best article selected from INSIDE INFO issues 13,14,15. Peter's article was an instructional guide to wiring up an extra keyboard to an Atari.

Of late, I have had the distinct feeling that INSIDE INFO is becoming increasingly technical in nature, going way above some of our members' heads. This guilt has prompted me to start up a beginners' column in the next issue. I would like to hear from members' who are new to computing on the Atari and find out in which direction the column should start and head.

Now for a few pieces of trivia. It has been announced by Atari that the new Atari 8 bit portable computer (65-XEP) will be scrapped. It is believed that the 65-XEP will be replaced by a more powerful portable version of the new Atari 16 bit machines. Smart move. If that wasn't enough, Atari also said that the operating systems for the ST computers will not be on-board. The ST operating system will be disk based rather than in ROM chips. So BASIC, LOGO and GEM will have to be loaded from a 3 inch floppy. Rumours are that Atari are having troubles with the operating system chips in a rush to get the ST's off the assembly line. One machine that is off the production line is the new 130XE. And talking of chips, the AMIE chip is troubling Atari causing delays in the 65XEM. Now for the good news?...

It is strongly rumoured that the new Australian distributor for Atari will be INDELEC Pty. Ltd. Please refrain from bothering INDELEC until their distributorship is finalised through advertising.

An upgrade from DOS 3.0 will soon be available in the form of DOS 2.5. It is upwardly compatible with DOS 2.0 and allows enhanced density with the 1050 drive.

Well, that's about all I have to say and I hope you enjoy this issue.

Ken Shiu

NOTICES

NEW MEETINGS VENUE

Due to a lot of trouble we encountered at our previous meetings' place, the committee decided to look for a new venue. Our new venue is in the 2nd Floor of the YWCA (Corner of Wentworth Avenue and Liverpool Street, adjacent to Hyde Park South). Meeting times are as usual and to settle the members down a trifle, coffee will be available at meetings for a mere twenty cents. A restaurant/canteen is open from 5PM to 7PM on the ground floor, offering hot and light meals.

NEW INFORMATION PAGE

As you may have noticed the Information Page has been moved to the second last page of INSIDE INFO. As you should be aware new membership is $20 and renwal $15. This can all be carried out on the tear-off slip at the bottom of the page. You should be constantly aware of updates on the Information Page in subsequent issues.

CLUB T-SHIRT PRINTS AVAILABLE

If you have a plain T-shirt lying around, you might like to jazz it up by having the ACE logo printed onto it. You can have this done at club meetings from now on. The styles available are:-

1. 8cm round logo, with lettering around border. $1.00
2. 7cm high x 14cm long logo with fancy lettering to right of logo (as on Inside Info). $2.00
3. 24cm round logo (no lettering). $4.00

These styles are available in BLACK, BLUE, RED , YELLOW or WHITE(on request).

Should there be members who are unable to attend meetings, an order can be placed by sending a note to Club's postal address, stating METRIC shirt size, COLOUR, and style. Please include cheque to cover cost of STYLE, T-SHIRT ($4.00) or SLOPPY JOE ($6.50) and Postage ($1.35) and your address.

SOFTWARE EXCHANGE NEWS

from Garry Francis

In INSIDE INFO No.17, I gave a broad outline for the reorganisation of the Software Exchange. After five months of hard yakka, I'm glad to say that this is virtually complete. Thank you all for waiting so patiently while things were sorted out. A summary of the current situation appears below.

Boxes of 10 blank Nashua disks are available for $27.00 plus postage. (Surface mail postage is currently $1.20 in N.S.W and $1.55 interstate.) These are very attractive, top quality, single-sided, double density disks with hub rings and there's no reason why you can't use the second side. Price includes a plastic box, labels, write protect tabs and a five year warranty!

Boxes of 10 blank C-45 cassettes are available for $9.00 plus postage. (Surface mail postage varies between $2.20 and $5.20 depending on the zone. Write for the exact cost or enquire at your local post office.)

There are now 11 disks of software in the Software Exchange including six brand new titles. This is all top-notch material, not to be missed by any self-respecting Atari user. The price of all disks has been reduced to $8.00 including surface mail postage anywhere within Australia.

INSIDE INFO Vol.1: This contains all the programs from INSIDE INFO issues 1-6. There are 42 BASIC programs and 4 assembly language source listings including a lotto selector, a GRAPHICS 0 colour selector, a keyboard organ, Mastermind, Snake, Squares, a flashing cursor routine, Doggies, a program listing utility, Grab An Apple, Boxes, a Greek alphabet and stacks of colourful graphics demos and tutorials.

INSIDE INFO Vol.2: This contains all the programs from INSIDE INFO issues 7-10. There are 21 BASIC programs and 2 assembly language source listings including fine scrolling demos, a fast block move using strings, menu programs, a cataloguing program, electronic poker, 7 mathematics programs and various sound and graphics demos. In addition to all that, there are 8 cartoons from Allan Smyth including Hagar the Horrible, Mickey Mouse, Andy Capp and more.

INSIDE INFO Vol.3: This contains all the programs from INSIDE INFO issues 11-15. There are 21 BASIC programs including a routine to set up the controller ports for input or output, a utility to ease the task of checking for typos, a simple terminal program, a sector reader, a light pen demo, a page flipping demo, a lotto selector that talks to you if you've got S.A.M., a sector editor, Gobbler, Tiny Text, graphs programs, character set utilities and more.

Atari XL Translator: This double sided disk allows XL owners to run software that could previously only be used on the 400/800 because the authors had broken Atari's rules and used illegal entry points into the Operating System. This is an essential item for all 600XL and 800XL users.

Team Atari fig-FORTH: In case you didn't know it, FORTH is one of the many powerful programming languages that can be used as an alternative to BASIC. This disk contains a full implementation of fig-FORTH (as defined by the Forth Interest Group) with a good editor and heaps of extensions to cater for the Atari's input/output, sound and graphics capabilities. Superb value!

Computer Cricket: This is K. J. Bricknell's monumental simulation of limited-over cricket as documented in INSIDE INFO No.17. One or two players can captain any of the six touring parties on the disk or you can use the utilities provided to modify them or create your own teams. Another utility allows you to play a whole series! This was added after the original article appeared in INSIDE INFO, but is documented on the disk.

Digitised Photos: This is an autobooting disk of four digitised photos which really show off the potential of graphics modes 9, 10 and 11. Play with the START, SELECT and OPTION keys to change pictures, colours and special effects. Very impressive, but two of the pictures are ... er ...R-rated?

Slide Show Vol.1: This is an autobooting, self-running slide show featuring ten full colour pictures of popular cartoon characters such as Wal and the dog from Footrot Flats, Snake, Charlie Brown, Fred Bassett and more. Four of these appeared on the front cover of INSIDE INFO No.17. The pictures were created by Ian Champ using the Atari Touch Tablet and Atari Artist (or Micro Illustrator) software, while the automation and fancy dissolves between pictures are from Joseph P. Grande's "Fader" in Antic September 1984. This is a great disk for showing off your Atari or for use as an in-store demo. If anyone has created their own high quality pictures (by any means whatsoever), please send them in for inclusion on future Slide Show disks.

Adventure Collection Vol.1: This is the first disk of Adventures prepared for the Adventure S.I.G. It contains four

Adventures from past issues of the now defunct SoftSide magazine. Three of these are traditional two-word text Adventures, while the fourth allows only single word input, but has illustrations of every room. The games are all reasonably easy and are therefore good introductory Adventures for beginners. (The next volume will be better and harder!) Incidentally, this disk has also got a very nice title page!

Page 6 Vol.1: Page 6 is a bi-monthly Atari users' magazine which is published in the U.K. It is a cross between the totally commercial magazines (like Antic and ANALOG) and the higher quality users' group magazines/newsletters (like INSIDE INFO). We have had an exchange subscription with Page 6 since the magazine's early days and now, after much typing of programs, correspondence with Page 6's editor and long distance disk swapping, I'm happy to present a complete collection of all their programs since the very first issue! I have prepared six disks and will "officially" release two with each issue of INSIDE INFO over the next few months (although you can order all six if you like). The first disk contains all the programs from issues 1-3. There are 21 BASIC programs including a Mastermind-type game, a line listing utility, Vultures, Four In A Row for two players, a routine to print playing cards in GRAPHICS 0, a calendar program, a neat maths program based around cricket, two character design utilities, a master directory utility and several graphics demos. (Incidentally, some issues of Page 6 can be borrowed from the library or you can subscribe by writing to Page 6 Magazine, P.O. Box 54, Stafford, ST16 1DR, U.K. 6 issues costs 10.50 pounds for surface mail or 16.00 pounds for airmail).

Page 6 Vol.2: This contains all the programs from Page 6 issues 4-6. There are 24 BASIC programs including a lunar lander, Merlin's magic square, a disk sorting utility, a finance program, Target, a tutorial on POKEing to screen memory, two items of music, vertical movement for player-missile graphics, a tutorial on binary numbers, Scramble, Dodger and lots of other neat stuff.

A PLEA FOR MORE SOFTWARE!

About 18 months ago, I started collecting and cataloguing software from all the computer magazines. The reason for this was threefold. Firstly, I wanted to get my own collection into some sort of order. Secondly, some of the magazine software is in the public domain and I wanted to add it to the Software Exchange for the benefit of all members. Thirdly, we needed some software to put on the Bulletin Board.

To keep a long story short, I am now in the position where I have a heap of magazine programs, but there are too many gaps in the collection to put it in the Software Exchange yet. We want a complete collection of every program ever published in a particular magazine before we release it. All you people out there can help by submitting all the programs you have ever typed in from magazines. Your disks will be returned as soon as possible, and for every program that we don't already have, you'll receive at least one in exchange. So it's a case of first in, best dressed. To make your job easier and eliminate some of the guesswork, I'll begin publishing lists of the programs that we need. This issue concentrates on Antic because it is the closest to being complete and most of their stuff is in the public domain. If you have typed in any of the following programs, please put them on a disk and give it to me at a meeting or post it to Software Exchange, Atari Computer Enthusiasts (N.S.W.), G.P.O. Box 4514, Sydney, N.S.W. 2001.

August 1982:
Translate
October/November 1982:
Bubble Sort (Assembly)
December 1982/January 1983:
String Search (BASIC and Assembly)
Grafix (2 listings)
Christmas Mailing Lister
February/March 1983:
Valentine's Day Card
Ut, Ut and Away (2 listings)
Grading on the Curve
April 1983:
Supercharging BASIC (BASIC and Assembly)
May 1983:
Invisible Sector (2nd program)
Idiotproofing Your Keyboard
June 1983:
Dancin' Man
Autoboot for BASIC (Assembly)
Stargazing
July 1983:
Conserve RAM
Talk is Cheap (Assembly)
Hokey Pokey Interrupts (Assembly)
August 1983:
Turn Coat
3-D Fuji (2 listings)
Micropainter Picture Utility
Color Codes for Resistors (2nd program)
September 1983:
P-M Tutorial
Pattern Maker
October 1983:
Reset to Rerun
Time Your Fire
Make A Face
Microloader (2 listings)
Mountain Skiing
Autoboot for Cassettes
Maxter Mind (Assembly)
November 1983:
Sound Demo
Here's Looking At You S.A.M. (Assembly)
Sound Beyond Sound (2 listings)
Simple Synthesizer
Tag
December 1983:

☞

More Interrupting (3 listings)
March 1984:
Atari's Cinderella (3 listings)
Flash Count
April 1984:
Control Your Environment
Electronic Birthday Card
Update Disks With Note and Point
June 1984:
Title Sampler
Scroll Your Way To The Top (2 BASIC and 1 Assembly)
Use BASIC To Animate
Color Finetuner (3 listings)
Shortcuts to Success (4 Assembly)
July 1984:
Break Key Shortcut
All Mixed Up
The Olympic Spirit
Interrupt with ACTION! (ACTION!)
Run Robot Run
Cassette Lookalike (Assembly)
August 1984:
Recall
It's Mere Horseplay
Cross Reference
Renamer
September 1984:
Three Color Demo
ANTIC's Mode 3 (4 listings)
Price's Color Picture Painter
Rotate Your Player (Assembly)
Customising BASIC Key Functions
Olympic Dash (2 listings)
Copy Mate (BASIC and Assembly)
October 1984:
Cruel I/O
Mole Attack
Plus Minus
ANTIC 4/5 Editor Animator (5 listings)
Jacks
Diver
November 1984:
GRAPHICS 9 Demo
ASCII Cursor Routine
Computer Brainwashing
Word Counting Program
First Lesson in Assembly Language
Adventure Island
Advent X-5
Windows (2 Assembly)
Create Your Own Cursor (Assembly)
December 1984:
Info Bits
Word Storage Space Saver (BASIC and Assembly)
Biffdrop
Windows Part II (BASIC and Assembly)
All About I/O (2 Assembly)

POWERPAD REVIEW

By Richard Taylor

The Powerpad is a tablet which allows highly detailed graphics to be created by tracing an image onto it. The PowerPad system can be either cartridge or disk based although you do NOT get both with the PowerPad. The PowerPad is very large, it measures 20" X 17" X 1.5" and an active surface area of 12" X 12". The Powerpad plugs into controller port 1 and is said to work on all Atari computers including the XL series.

To do anything with the PowerPad you need to purchase one of the many programs avaliable. I chose the "Micro Illustrator" which is a drawing program for the PowerPad. Upon unpacking Micro Illustrator from its box you find:- a stylus, an instruction book and the Micro Illustrator cartridge.

THE MAIN MENU

After you have powered up the system you are confronted by the Title page and then the Main Menu. This consists of ten drawing functions: Draw, Point, Line, Lines, Rays, Fill, Frame, Box, Circle, and Disk. The difference between Frame and Box is that Frame draws a 'wire' frame and Box draws a solid box. The same applies to Circle and Disk. The rest of the functions are self explanatory. The menu also contains six utility commands: Erase, Storage, Mirror, Magnify, Scaled, and Color menu. Further down the screen there are 10 brush sizes that range from a single pixel to an open diamond and square.

At the bottom of the screen the background color and three main colors are appear. There are twelve secondary or 'checkerboard' colors beneath this. There are three secondary menus that the user arrives at by simply pressing the Erase, Storage, or Color menu on the screen.

THE ERASE MENU:- When you press the erase 'button' the screen changes to a very simple self explanatory message: Touch Yes or Y to erase, anything else cancels.

THE STORAGE MENU:- When you press the storage button the screen changes to Storage and five options on the screen, they are: Load Picture, Save Picture, Index Disk Pictures, Delete Disk Pictures, and Format Blank Disk. Despite the fact that most of the options are disk oriented the Load and Save options can also be used with cassette tape.

THE COLOUR MENU:- This is split into three parts. On the left side of the screen are selection directions and four options that you can choose from: Rainbow Color, Default colors, Restore Last Color, and Adjust Colors on Picture. The right side of the screen contains two vertical bars. The bar to the left (color) represents the 16 colour options (hues) of the Atari computer. The bar to the right (level) represents the sixteen levels of luminence (brightness) of each of the colors. Across the bottom of the menu are the current main and checkerboard colors. The three default main colors are Orange, Green, and Blue. The default background color is Black.

LIMITATIONS

A limitation is that there is no option to print the pictures you create and as I recently purchased an Atari colour printer this is a big disappointment.

Another feature that would have been useful would be the ability to insert text on the picture screen.

The PowerPad is capable of reading more than one point on its surface at a time however this program does not make use of this feature.

All software that you purchase for the PowerPad comes with its own plastic overlay that clips onto the surface of the PowerPad. These overlays vary depending on the program being used.

For the Micro Illustrator the overlay is a transparent grid-sheet with five buttons drawn at the top. The buttons are, from left to right: Pen Down, Menu, Pen Lock, and Pen Down. The buttons are repeated on each side so that the program can be used by both left and right handed people.

SUMMARY

The PowerPad was a good purchase, however if you are thinking of buying one and want it demonstrated, have the demonstration done on an Atari and not a Commodore or an Apple as each program varies from computer to computer.

Stay tuned for a review on the PowerPad programming Kit.

AN ALTERNATIVE HANDLER FOR THE ATARI 1020 PRINTER

by David and Mary Lynch.
Adapted from Page 6, Issue 13.

[This is a handy little program that shows how you can modify a device handler to satisfy your needs. - Roberto R.]

One of the first ways in which you can use your printer is to list your own programs.

For the ATARI 1020 printer it is much more convenient to use the small character size in order to obtain 80 characters per line. However, this involves having to use CTRL S every time the small characters are desired.

Our first attempt, successful but cumbersome, was to write the listing file to disk and then use a program to read in the file and write it to the printer, preceded by esc-ctl-S. This involved reading and writing large files - slow. We tried various other tacks, including esc-ctrl-S as a comment in the program and using a vertical blank interrupt routine to write esc-ctrl-S, until we finally came up with a workable solution.

The problem is that when confronted with a command LIST"P:" to print a listing the Operating System first closes then re-opens the printer, thus setting default values and defeating any chicanery you may have been up to. The solution was to define a new input handler "L:" whose only function is to print in small letters. We steal most of the code provided by the Operating System for the printer, "P:", but substitute our own code for the OPEN routine. When we set up "L:", we open the printer then pint esc-ctl-S. (Note that because "P:" and "L:" share code and buffers, it is dangerous to have them OPEN at the same time - not that I can see any reason fo doing so.)

There are a couple of additional frills. How about being able to forget about setting up "L:" each time we turn on the computer ? We can do this by making the program into an AUTORUN.SYS file which is executed automatically by DOS on power-up. In addition, we do not want our facility ruined by pressing SYSTEM RESET and we can stop this by placing the address of the set up routine in RUNAD ($2E0,$2E1), which is executed when SYSTEM RESET is pressed.

For those with an assembler, a listing of the assembly language code is provided. Save the object code as "D:AUTORUN.SYS". The printout is of the assembler listing rather than the source so that the BASIC program may be more understandable.

The program occupies the end of page 6, $6A0 to $6FF. If you don't mind SYSTEM RESET destroying "L:", $6DC to $6FF is all that needs preserving.

The BASIC program does the same thing. Type in the program as listed. SAVE it as "L.BAS" and the RUN the progam with a DOS disk in the drive. Turn off the computer and turn it on again. If all has gone well you should now have set up "L:". Try LOAD "L.BAS" then LIST"L:" and you should get a listing of the program in letters 80 characters to a line.

LPRINT does not appear to work very well with the 1020. While you can print 40 characters per line, the special facilities, such as changing print size, do not work. I suspect that this is because when BASIC meets an LPRINT statement, it closes and opens the printer, so destroying control commands that you have sent it previously.

The solution is to amend programs containing LPRINT as follows:
(a). Precede the first LPRINT statement executed in the program by CLOSE#7:OPEN #7,8,0,"P:"
(b). Replace all LPRINT's by PRINT#7;

```
1 REM ################################
2 REM #     L: PRINTER HANDLER       #
3 REM #   by David and Mary Lynch    #
4 REM #    Reprinted from Page 6     #
5 REM #          Issue 13.           #
6 REM # Published by Atari Computer  #
7 REM #    Enthusiasts (N.S.W.)      #
8 REM #          June 1985           #
9 REM ################################
1030 OPEN #2,8,0,"D:AUTORUN.SYS"
1040 TRAP 1060:FOR I=1 TO 500
1050 READ Q:PUT #2,Q:? Q:NEXT I
1060 END
1070 DATA 255,255,160,6,255,6,165
1080 DATA 12,141,182,6,165,13,141
1090 DATA 183,6,169,181,133,12,169
1100 DATA 6,133,13,56,176,3,32,184
1110 DATA 6,160,0,185,26,3,201,0,240
1120 DATA 9,200,200,200,192,34,208
1130 DATA 242,56,96,169,76,153,26
1140 DATA 3,200,169,220,153,26,3,200
1150 DATA 169,6,153,26,3,96,234,6
1160 DATA 219,238,157,238,166,238
1170 DATA 128,238,157,238,76,120,238
1180 DATA 32,159,238,169,27,32,167
1190 DATA 238,169,19,32,167,238,169
1200 DATA 155,32,167,238,160,1,96
1210 DATA 224,2,225,2,160,6
```

```
10 ;##################################
20 ;# L:PRINTER HANDLER SOURCE CODE #
30 ;#    by David and Mary Lynch    #
40 ;#     Reprinted from PAGE 6     #
50 ;#           Issue 13            #
60 ;#  Published by Atari Computer  #
70 ;#      Enthusiasts (N.S.W.)     #
80 ;#           June 1985           #
90 ;##################################
0100 ;
0110     .OPT NO EJECT
0120 ;
0130     *=  $06A0
0140 ;
0150 HATABS = $031A
0160 DOSINI = $0C
0170 ;
0180 ;  System Reset trap
0190 ;
0200 LINIT LDA DOSINI
0210     STA SYSRES+1
0220     LDA DOSINI+1
0230     STA SYSRES+2
0240     LDA # <SYSRES
0250     STA DOSINI
0260     LDA # >SYSRES
0270     STA DOSINI+1
0280     SEC
0290     BCS LSETUP
0300 ;
0310 SYSRES JSR DUMMY
0320 DUMMY
0330 ;
0340 ; look for free handler entry
0350 ;
0360 LSETUP LDY #0
0370 LOOP LDA HATABS,Y
0380     CMP #0      ; free entry?
0390     BEQ FREE
0400     INY
0410     INY
0420     INY
0430     CPY #34     ; end of HATABS?
0440     BNE LOOP
0450     SEC         ; error,no free entries
0460     RTS
0470 ;
0480 FREE LDA #'L    ; device name
0490     STA HATABS,Y
0500     INY
0510     LDA # <LTAB
0520     STA HATABS,Y ; handler address
0530     INY
0540     LDA # >LTAB
0550     STA HATABS,Y ; handler address
0560     RTS
0570 ;
0580 LTAB .WORD LOPEN-1 ; new open
0590     .WORD $EEDB ; old close
0600     .WORD $EE9D ; old read
0610     .WORD $EEA6 ; old write
0620     .WORD $EEA0 ; old status
0630     .WORD $EE9D ; old special
0640     JMP $EE78   ; old init
0650 ;
0660 LOPEN JSR $EE9F
0670     LDA #27     ; escape
0680     JSR $EEA7
0690     LDA #19     ; ^S
0700     JSR $EEA7
0710     LDA #155    ; CR
0720     JSR $EEA7
0730     LDY #1
0740     RTS
0750 ;
0760     *=  $02E0
0770     .WORD LINIT
0780     .END
```

DOCTOR DOS

by Garry Francis

There are very few Atari experts in Australia, so it's hard to find someone to turn to when you've got a programming problem. With this in mind, I've decided to start a "question and answer" column, but I can't afford the time to write it indefinitely. If there are any Atari experts out there who feel qualified to take over from me, then please let me know. In the meantime, if you need any help, just write to DOCTOR DOS, c/- Atari Computer Enthusiasts (N.S.W.), G.P.O. Box 4514, Sydney, N.S.W. 2001. Your question will be printed in INSIDE INFO and my successor or I will do our best to provide an answer.

* * * * *

ELIMINATING SCREEN GARBAGE

Dear Doctor,
I typed in Wildwest and Courting Crickets from INSIDE INFO No.16. Both programs work perfectly, but there is some "screen garbage" during initialisation. Is there a way to eliminate this by blanking the screen until the game is set up? - Danny McClatchey

Yes. The easiest way to eliminate screen garbage is to disable DMA (Direct Memory Access) early in the initialisation routine, then re-enable it when initialisation is complete. This can be done with two simple POKEs. To disable DMA, use POKE 559,0. To re-enable it, use POKE 559,X. Both these POKEs should be carried out after any GRAPHICS commands as the GRAPHICS command will itself enable DMA. The value of X in the second POKE is an 8-bit integer value (i.e. a number from 0 to 255) which varies depending on the application. I haven't got enough room to discuss it here, so refer to the hardware manual (there's a copy in the library) or an article on player-missile graphics for further details. If you are only using the built-in graphics modes, then use POKE 559,34. This sets bits 2 and 5 to enable DMA with a standard width playfield.

Wildwest can be cleaned up by adding two POKEs. Firstly, add POKE 559,0 to the beginning of line 140. This disables DMA while the subroutine at line 450 is executed. When the subroutine is finished and returns to line 140, DMA is automatically re-enabled by the GRAPHICS 0 command. Secondly, add POKE 559,0 to the beginning of line 170 or 580 (it doesn't matter which). This disables DMA while the custom display list is set up. DMA is re-enabled by the POKE 559,46 in line 210.

In Courting Crickets, Stan has remembered to disable DMA while various things are set up, but some of the POKEs are in the wrong place to be most effective. To fix this, add POKE 559,0 after the GRAPHICS 0 in line 350 and delete the other two POKE 559's from the same line. DMA is re-enabled by the POKE 559,62 in line 1740. There is still a brief flash of garbage at one point during the initialisation, but I couldn't figure out how to get rid of this.

* * * * *

SHUFFLING ROUTINE

Dear Doctor,
How do I write a loop that picks a series of random numbers from a limited range such that none of the numbers are repeated? I want to use it in a program to pick Lotto numbers. - Vaughn De Vocht

What you really need is a shuffling routine. Here is a routine that is reasonably fast, efficient, easy to understand and most importantly, every possible combination has an equal probability of occurring. It is split into two parts. The first part simply initialises variables and sets all the elements of the array to their initial (ordered) values. This only needs to be done once at the beginning of the program. To alter the upper limit of the range of numbers, just alter the value of MAX. For example, MAX=40 to shuffle your Lotto numbers, MAX=52 to shuffle a deck of cards or MAX=100 to determine the order that questions will be asked in a 100 question trivia quiz. N determines how many numbers to choose from the set and must be less than or equal to MAX. For example, N=6 to pick 6 Lotto numbers, N=5 to pick 5 cards for a single hand of poker or N=100 to shuffle the entire list of questions in the trivia quiz.

```
10 GRAPHICS 0:MAX=40:N=6:DIM A(MAX):FOR I=1 TO MAX:A(I)=I:NEXT I
```

The second part is the shuffling routine. It can be carried out any number of times without re-initialising the array.

```
20 FOR I=1 TO N:J=INT(RND(I)*(MAX+1-I))+I:TEMP=A(I):A(I)=A(J):A(J)=TEMP:NEXT I
```

This is all you need to incorporate a shuffling routine in your own program. If you want a demo, then add the following few lines to print the shuffled numbers (though this could also be done within the shuffling loop) and allow the user to go back and pick some more numbers. Hey presto! A full-fledged Lotto selector in just 6 lines of code!

```
30 FOR I=1 TO N:PRINT A(I);" ";:NEXT I
40 PRINT :PRINT :PRINT "PRESS ANY OTHER KEY TO PICK MORE NUMBERS":POKE 764,255
50 IF PEEK(764)=255 THEN 50
60 PRINT :GOTO 20
```

* * * * *

COPYRIGHTING SOFTWARE

Dear Doctor,
How does one go about copyrighting a program and also a company's name? - Richard Taylor

(a) The first copyright legislation in Australia was the Copyright Act of 1912 which was the same as the British Copyright Act of 1911. This gave copyright protection throughout Australia and the British Empire and was further widened when Australia joined the Berne Union. A new Copyright Act passed in 1968 introduced new features which allowed Australia to join the Universal Copyright Convention and therefore widen international recognition to include U.S.A. and U.S.S.R. - neither of which were in the Berne Union.

The Copyright Act (1968-73) provided for copyright in two areas, mainly "works" and "subject matter other than works". Works were defined as "literary, dramatic, musical or artistic work". Following the recent case between Apple and Computer Edge, the Full Federal Court decided on 29th May, 1984 that computer programs in source code were indeed "literary works" and programs in object code were "translations" of their corresponding source codes. As such, both source and object code were protected by the Copyright Act. The Copyright Amendment Act 1984 was introduced into parliament a week after the Full Federal Court decision. This contained express reference to computer programs in the definition of literary works and the definition of computer program was itself expressed so as to include both source and object code.

In Australia, copyright exists from the time an idea is reduced to writing or some other material form (including typing of programs directly into a computer's memory). It gives the creator the sole right to reproduce, publish or adapt the work. This right stays with him throughout his life and passes to his heirs for 50 years after his death. The creator or his heirs can sell some or all of these rights at any time. Note that you do not have to register a work to gain copyright in Australia. However, you may improve your protection in other countries by registering it in those countries (if appropriate) and marking it with the word "COPYRIGHT" or the copyright symbol (a small C within a circle), followed by your name and the year of publication. In the case of computer programs, I'd suggest putting this in your documentation, at the beginning of your source code and also having it appear on the title screen of your program when it is run. This at least makes people aware that it is not in the public domain.

If someone infringes your copyright, you can sue for damages.

(b) If you want to set up your own business, there are essentially three options open to you. Firstly, if operating by yourself, you can become a sole trader and use your own name (e.g. John Smith, J. Smith or Smith) without having to register the name. Secondly, if two or more people are involved, you can form a partnership and again use your own names (e.g. Smith, Jones and Williamson) without having to register. However, if either type of business wishes to use a name other than their own (e.g. Smith's Software or Jones and partners or Megadata), the name must be registered with the Corporate Affairs Commission (or its equivalent in other states). Thirdly, you can form a company which must be registered with the Corporate Affairs Commission before it can begin trading. A company only has to register its business name if it is different to the name under which it is registered.

There are many restrictions on the choice of business name. For example, it cannot be similar to an existing firm, it must not be misleading, it cannot use the words "imperial", "commonwealth" or "international" and it must not suggest a connection with the Crown, the Royal Family, etc. When the name is approved, you will be issued with a certificate of registration which must be prominently displayed at your place of business.

(c) I believe this information to be reasonably accurate at the time of writing, but I'm a software expert, not a lawyer. Make sure you confirm anything I've said before taking any action. The Australian Copyright Council can answer questions on copyright and the Corporate Affairs Commission can answer questions on registering a business name. Note that you do not copyright a business name.

* * * * *

DISASSEMBLING A BINARY FILE

Dear Doctor,
I believe that once you have assembled source code, there is no way back. Is this true? If it's not true, then can

the Atari Assembler Editor disassemble a binary file or is there another disassembler that can? - Guy Bao

No, this is not true. It is fairly easy to disassemble a binary file. In the case of the Assembler Editor cartridge, you must first load the binary file into memory using LOAD#D:FILENAME.EXT then proceed according to the instructions on page 38-41 of the Assembler Editor Manual. Needless to say, you must know the starting address of the binary file so you know where to start the disassembly. If you're not sure what it is, then use the following BASIC program to find it for you. Just replace the filename in line 10 with your own filename.

```
10 TRAP 50:GRAPHICS 0:OPEN #1,4,0,"D:FILENAME.EXT"
20 GET #1,LO:GET #1,HI:IF LO<>255 AND HI<>255 THEN ? "NOT A BINARY FILE":GOTO 50
30 GET #1,LO:GET #1,HI:? "DECIMAL STARTING ADDRESS=";LO+256*HI
40 GET #1,LO:GET #1,HI:? "DECIMAL ENDING ADDRESS=";LO+256*HI
50 CLOSE #1:END
```

There are also dozens of other disassemblers available. Virtually every assembler, machine language debugger, monitor and disk editor has a disassembler built in. You can also buy a dedicated disassembler or type one in from a book or magazine. Disassemblers have appeared in Compute! March 1981 and June 1981, Kilobaud Microcomputing August 1981, Antic April 1983, ANALOG #14 November/December 1983, "Machine Language for Beginners" by Richard Mansfield (Compute! Books) and "Beyond Games: Systems Software for Your 6502 Personal Computer" by Ken Skier (Byte Books). The best disassembler on the market is "Ultra Disassembler" from Adventure International. This produces the most readable code, it uses the official Atari labels for all Operating System equates and it can immediately be re-assembled without having to make any changes! This is the one I use and recommend to serious assembly language programmers.

* * * * *

ATARI MODEMS & TERMINAL PROGRAMS

Dear Doctor,

Could you give me some advice on Atari compatible modems and terminal programs. I'd specifically like to know whether I need to buy an 850 Interface Module to use a modem and whether the Telelink II cartridge is any good. - Nigel Delaforce

First and foremost, you must appreciate that there are two telecommunications standards used throughout the western world. Most of the world (including Australia) uses the CCITT standard, but the Yanks have got to be different (as usual) and they use the Bell standard. If you're going to buy a modem to communicate with other computers within Australia, make sure it conforms to the CCITT standard. This immediately eliminates all the modems I've ever seen advertised for the Atari in the U.S. magazines.

There are three ways of interfacing a modem to an Atari - through the RS232 ports on the 850 Interface Module, directly through the serial port or through one of the joystick ports. In the first case, you can use just about any modem on the market that has an RS232 port and uses the CCITT standard, but you will probably need to make your own cable. In the second and third cases, the modem must be made specifically for the Atari. Don't buy any of the ones advertised in the U.S.A. because they only support the Bell standard and can't be used to communicate within Australia. The Sunsoft modem reviewed in INSIDE INFO No.18 is an Australian designed modem which plugs directly into the serial port. This looks like an excellent product, but it only operates at 300 baud. I don't know of any Australian modems that use the joystick ports, but there may be some made in the U.K. or Europe which you can use in Australia.

As far as software goes, if your modem hooks up to the 850 Interface Module or the serial port, you can use just about any commercial terminal program or any of the many excellent public domain terminal programs. If your modem uses the joystick ports, you are restricted to the terminal software supplied by the manufacturer of the modem.

Telelink I and II only work with a modem hooked up to the 850 Interface Module and possibly the serial port. I wouldn't recommend Telelink I because it is a very "dumb" program and doesn't allow uploading and downloading of software. I haven't seen Telelink II in action, but for the money, I feel you'd be better off with one of the public domain programs.

In summary, I'd recommend either the Sunsoft modem (which includes software) or an 850 Interface Module, an Australian designed and manufactured modem (such as Sendata, Cicada, Avtek, Dick Smith, etc.) and a public domain terminal program (such as JTERM, AMODEM, etc.).

MODIFYING VISICALC

Dear Doctor,

I would like to reconfigure the keyboard when using Visicalc so that I don't have to hold down the control key to

use the cursor controls. I know that it is possible to reconfigure the keyboard, but Visicalc is an autoboot program that is loaded by turning the computer off then on. This reverts everything to the default state cancelling any keyboard configuration that might have been done. What gives me a slight glimmer of hope is that the XL translator disk does something to the system then allows an apparent cold start by pressing the SELECT key. Perhaps a program could be written to reconfigure the cursor control keys, then allow Visicalc to be booted by pressing the SELECT (or some other) key.
(a) Is there a way to do this?
(b) If it's possible, could anyone out there write the software for me? I'm useless at anything other than simple BASIC and I've never used machine language.
(c) Could the plain blue lettering and background colours in Visicalc be changed at the same time?
- Peter Douglas

(a) Wow! That's quite a tall order! I'd suggest you write to the Visicalc people before trying anything yourself. You CAN do it yourself, but is it worth the trouble? Let's discuss the pros and cons.

Firstly, you should establish whether redefining the keys is a wise move. I personally think you're making things more difficult for yourself because you will still be using the CTRL key combinations in other environments like word processing, programming or just typing in a BASIC program listing out of a magazine. This brings about a psychological conflict as your brain tries to adapt to the different environments. The end result is a greater number of mistakes brought about by incorrect keystrokes. I experience this all the time when I transfer between my Atari and the mainframes at work, or even between different terminals with slightly different keyboard layouts. You will also have to reconfigure the +, -, * and = keys which will further add to the confusion.

The idea of configuring the keyboard, then running Visicalc in a similar fashion to the way the XL translator works is an interesting one. It is certainly feasible (even on a 400/800), but you probably won't get it to work. The reason is that Visicalc will disable anything that you've done when it does its own initialisation. You will therefore have to modify Visicalc itself!

If you're still convinced to persue the matter, you must establish how Visicalc sets up its keyboard routines. It could be doing this on a number of levels. It could redefine the keyboard matrix, use its own maskable interrupt handler, use its own keyboard handler or use the inbuilt keyboard handler and simply modify its actions depending on which key is pressed. The only way to find out what technique (or combination of techniques) is being used is to disassemble the entire program. This requires a disassembler capable of disassembling directly from disk sectors. It is also preferable that the disassembled source code uses the standard Atari Operating System equates, otherwise you'll never find your way around. Adventure International's "Ultra Disassembler" mentioned above is ideal. Needless to say, you'll also need a sound knowledge of assembly language, the Atari Operating System and the structure of a boot disk.

Once you've got a disassembled listing of Visicalc, you'll need to spend days and days wading through the code trying to find the keyboard routines and then trying to understand how they work. This is a nightmare! You can then modify the routines (or more likely you'll have to add your own, but figuring out where to put them is another nightmare) and re-assemble the whole lot. Needless to say, this last operation requires an assembler. The MAC/65 supercartridge from O.S.S. is undoubtedly the best and costs $99.95. Once it is tested and debugged, you can write a utility to transfer your new object code to a boot disk and you're pretty well ready to go. Simple huh?

But what about if Visicalc has copy protection? What about if the object code is encoded or scrambled or something? What about if you have a weak heart?

(b) Is anyone prepared to follow up on question (b)? I've let you know what's involved, but I certainly wouldn't tackle it myself unless I was being paid by the hour!

(c) No worries. Once the program is disassembled, you can patch in the extra 5 bytes of code to change the background colour and a further 5 bytes if you wish to change the luminance of the lettering. This could go immediately after the code to open the screen.

* * * * *

USING ATARI WORD PROCESSOR WITH THE 1020 PRINTER

Dear Doctor,
I have the Atari Word Processor, which is a disk-based package containing two diskettes, a data diskette and an extensive tutorial-type manual. It's brand new and has only been tested to prove it works. I want to sell it and a friend of mine with an 800XL, 1050 disk drive and 1020 colour printer wants to buy it. The problem is that the word processor is an autoboot disk and won't load unless the system contains an 850 Interface Module, which my friend

doesn't have and doesn't need and doesn't want to buy. Mine isn't for sale. Can anyone think up a way to get around this problem, such as reconfiguring the printer handler then allowing a "cold start" as described above? Alternatively, I could sell the disk-based word processor to someone out there who has an 850 Interface Module ($90 O.N.O.) and get a simpler word processor such as AtariWriter for my friend. Along these lines, is anyone with an AtariWriter willing to sell it? - Peter Douglas

This is a mystery to me, because I'm not aquainted with the 1020 colour printer and I know even less about the Atari Word Processor than I do about Visicalc. (Incidentally, I don't own either.) If any readers can help, please contact Peter on (067) 72-1183 or (067) 72-6315 or write to P.O. Box 1000, Armidale, N.S.W. 2350.

* * * * *

".GOING FORTH"

By Bob Vickery

FORTH is the most powerful language available for the Atari computer but very few members of A.C.E. have developed an interest in it. This situation may be due, in part, to a shortage of documentation. The "Team Atari fig-FORTH" disk distributed by A.C.E. comes with very little back-up and the standard book on FORTH describes a dialect which is different from fig-FORTH. I thought it would be worth-while to list some of the help available to would-be FORTH users.

The nearest source of help is the Sydney Chapter of the Forth Interest Group (fig). It meets on the second Friday of each month in Rm. L619 of the John Goodsell Building at the University of N.S.W., Kensington, and publishes a newsletter called "Forthwright". The group can be contacted through the secretary, Peter Tregeagle, 10 Binda Rd., Yowie Bay. Membership of the original Forth Interest Group costs $US33 per annum including air-mail postage of their journal "Forth Dimensions". The address is P.O. Box 8231, San Jose, CA 95155, U.S.A.

The cheapest version of the language is the "Team Atari fig-FORTH" disk available from A.C.E. for $10. This system adheres closely to the FORTH-79 standard with extensions to handle the special features of the Atari such as sound and player-missile graphics. It has many good things, but some deficiencies. The first of these is lack of documentation as mentioned before. The second is a tendency to crash or to hang up unexpectedly. A bug-free, well-documented version of FORTH-79 can be bought from Valpar International, 3801 East 34th Street, Suite 105, Tucson, Arizona 85713, U.S.A. Valpar also have 6 disks of extensions to the language including a DOS system.

fig has released the new FORTH-83 Standard but this has not been implemented on the Atari, as far as I know.

There are two books which I would recommend to any FORTH programmer. Firstly, there is the "fig-FORTH Installation Manual" available from fig and also in the A.C.E. library. The second is "Starting Forth" by Leo Brodie which is in most stores selling computer books. The Manual has a complete description of all the essential words in the FORTH-79 kernal and all the code needed to install it in a certain (non-Atari) 6502 machine. It is also very, very heavy reading and not suitable for an introduction to .the language. This is where Brodie's book is outstanding. It takes the reader right from 2 2 + through to methods for modifying the compiler in easy stages with cute illustrations. Unfortunately, the dialect of FORTH described by Brodie differs in quite a few respects from FORTH-79 (and FORTH-83 too).

Here are some of the differences between the languages. The first of each pair of words is Brodie's version and the second is the Team Atari version.

' <> '

Brodie's ' ccc returns the code field address of the word ccc, whereas the Team Atari FORTH word ' returns the parameter field address. The sequence " ' ccc CFA " will return the code field address of the word ccc. ' is immediate and can be used in a colon definition.

'S = SP@

+LOOP <> +LOOP

In Team Atari FORTH +LOOP adds a signed increment to the index and compares it with the limit. The branch back to DO is taken if the new index is less than the limit (for positive increments) or greater than the limit (for negative increments).

2-	not implemented
2/	not implemented
2DROP	not implemented
2DUP	not implemented
2OVER	not implemented

2ROT not implemented
2SWAP not implemented
>IN = IN
?DUP = -DUP
?STACK <> ?STACK

In Team Atari FORTH this word automatically aborts if the stack is out of bounds.

ABORT" not implemented
BLANK = BLANKS
CREATE = <BUILDS

Team Atari FORTH uses the pair <BUILDS .. DOES> instead of CREATE ..DOES>.

DNEGATE = DMINUS
EMPTY not implemented
EXIT = ;S
H = DP
NEGATE = MINUS
NUMBER <> NUMBER

Team Atari FORTH always converts a string to a double length (32 bits) number. The position of the decimal point in the string is given in the User variable DPL.

PAGE = HEX 7B EMIT
U/MOD = U/
VARIABLE <> VARIABLE

The Team Atari FORTH word takes an initialization value from the stack.

WIPE = CLEAR

WHERE'S MY FILE?

By Joe Delman

After reading DISKussion in the past few INSIDE INFOs, I spent quite some hours playing with the layouts of my disks. For different exotic reasons, I sometimes needed to know on which sectors of a disk a particular file resided on. Doing this manually hurt my brain, so I wrote the program below to do this job for me.

Just RUN it and answer with the file's name - you can enter it as it appears in the disk directory, or as you would when LOADing the file. The program works for any file that you can duplicate using DOS 2.0 (well all the ones I've tested it on so far).

This program can be used on its own or as a subroutine. One idea I've thought of is to change a disk dissassembler I have which works on up to 300 consecutive sectors. You must supply the 1st sector and the number of sectors. As it's written in BASIC, it could easily be made more versatile by being able to dissassemble sectors on a file basis.

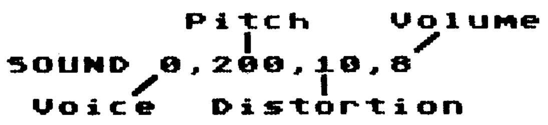
```
1 REM **********************************
2 REM *       FILE SECTOR LISTER       *
3 REM *          by Joe Delman         *
4 REM * Published by Atari Computer #
5 REM *     Enthusiasts (N.S.W.)       *
6 REM *            June 1985           *
7 REM **********************************
10 CLR :TRAP 10:POKE 82,2:POKE 1536,10
4:POKE 1537,32:POKE 1538,83:POKE 1539,
228:POKE 1540,96:POKE 769,1:POKE 770,8
2
20 DIM A$(128):POKE 773,INT(ADR(A$)/25
6):POKE 772,ADR(A$)-256*PEEK(773);DIM
FILE$(12),TEMP$(11)
30 TEMP$="           ":FILE$="
    ":? "INPUT FILENAME ";:INPUT FILE$
:SECT=361:J=-1:GOSUB 2000
40 GOSUB 1000:I=0
50 J=J+1:IF A$(I+6,I+16)=TEMP$ THEN 90
60 IF I<112 THEN I=I+16:GOTO 50
70 IF SECT<368 THEN SECT=SECT+1:GOTO 4
0
80 ? :? "FILE NOT FOUND!":? :? :GOTO 3
0
90 ? :? "FILE FOUND, USES ";ASC(A$(I+3
,I+3))*256+ASC(A$(I+2,I+2));" SECTORS"
:? :SECT=ASC(A$(I+5,I+5))*256+ASC(A$(I
+4,I+4))
100 J=J*4:N=1:STAT=ASC(A$(I+1,I+1)):IF
 SECT=0 OR STAT>98 OR STAT<32 THEN ? :
? "FILE DELETED, OPEN, OR HAS 0 SIZE":
? :? :GOTO 30
110 IF SECT=0 THEN ? :? :GOTO 30
120 ? "PROGRAM SECTOR ";N;" IS DISK SE
CTOR ";SECT:GOSUB 1000
130 N=N+1:SECT=ASC(A$(126,126))-J:SECT
=SECT*256+ASC(A$(127,127)):GOTO 110
1000 A$(1)=" ":A$(128)=" ":A$(2)=A$:PO
KE 779,INT(SECT/256):POKE 778,SECT-256
*PEEK(779):X=USR(1536):RETURN
2000 IF LEN(FILE$)=0 THEN RETURN
2010 L=0:FOR K=1 TO LEN(FILE$)
2020 IF FILE$(K,K)="." THEN L=K
2030 NEXT K:IF LEN(FILE$)=L OR L<2 OR
L>9 THEN RETURN
2040 TEMP$(1,L-1)=FILE$(1,L-1):TEMP$(9
,11)=FILE$(L+1,LEN(FILE$)):RETURN
```

☆☆☆

SOUND EFFECTS

by Andrew Hobson

Sound on the Atari is very simple -it does not need complex pokes or machine language subroutines as on other computers. All that is involved is the SOUND command. The SOUND command tells the computer which voice to use, the pitch, distortion and volume. Each SOUND command therefore must include four numbers to represent these four elements.



VOICE - The Atari has four channels. It is therefore possible to play four different sound simultaneously. The voice numbers are 0,1,2 and 3. Each voice needs a separate SOUND statement to play a note.
PITCH - This number indicates the frequency of the sound you require. The number must be between 0 and 255 - the higher the number the lower the note. e.g. 255 is the lowest note. The range is around 3 octaves.
DISTORTION - This tells the computer how 'pure' the note should be (0 is a pure tone while 8 is a buzz. The distortion should be an even number between 0 and 14).
VOLUME - This sets the volume of the desired voice (not all the voices). Volume ranges from 0 (silent) to 15 (loudest). Eight is the normal volume setting. If the added volume of the different voices is greater than 32 the sound may not be clear.
To turn a sound off, you must zero out all the voice parameters:-

SOUND 0,0,0,0

SOUND EFFECTS

To make interesting sound effects you need to use FOR/NEXT loops. You use these just the same as in ordinary programs to change the pitch.

```
10 FOR PITCH=100 TO 200
20 SOUND 0,PITCH,10,8
30 NEXT PITCH
```

Other sound effects can be produced from this program by changing the volume, distortion etc. and adding more loops. Usually the distortion is not changed, just the pitch and the volume.

```
10 FOR TUNE=1 TO 5
20 FOR PITCH=60 TO 0 STEP -1
30 SOUND 0,PITCH,10,15-PITCH/4
40 NEXT PITCH
50 SOUND 0,0,0,0
60 NEXT TUNE
```

It is possible to use a FOR/NEXT loop while changing the voices. When you do this the first voice you have set will continue playing up to and after you set the second (When you have voice 0 playing and you then set voice 2, voice 0 will continue playing.

```
10 FOR PITCH=254 TO 0 STEP -1
20 FOR VOICE=0 TO 3
30 SOUND VOICE,PITCH+VOICE,10,8
40 NEXT VOICE
50 NEXT PITCH
```

To be able to play a musical piece you need to use DATA statements and the READ command to gather the different pitches. Along with the pitch values should be a duration value to control the length of the FOR/NEXT loop.

```
10 FOR TUNE=1 TO 17
20 READ PITCH,TIME
30 SOUND 0,PITCH,10,8
40 FOR TIME2=0 TO TIME*50:NEXT TIME2
50 SOUND 0,0,0,0
60 NEXT TUNE
100 DATA 60,2,91,2,91,1,91,1,91,2,91,1
,81,1,72,1,64,1,60,4,60,2,81,2,81,2,64
,2,72,2,81,2,91,4
```

I hope this will make SOUND statements clear and easy to use. Following are nine short programs to demonstrate these effects on a range of topics. These can easily be merged into your own programs to add a touch of class.

```
10 FOR TUNE=1 TO 10
20 FOR VOLUME=14 TO 0 STEP -5
30 FOR PITCH=255 TO 0 STEP -10
40 SOUND 0,PITCH,10,VOLUME
50 NEXT PITCH
60 NEXT VOLUME
70 NEXT TUNE
```

```
10 REM **STEAM ENGINE**
20 FOR TIME=1 TO 10:NEXT TIME
30 SOUND 0,6,8,10
40 FOR TIME=0 TO 20:NEXT TIME
50 SOUND 0,2,3,6
60 FOR TIME=0 TO 30:NEXT TIME
70 GOTO 20
```

```
10 REM ** SIREN **
20 FOR TUNE=1 TO 50
30 FOR PITCH=60 TO 40 STEP -2
40 SOUND 0,PITCH,10,8
50 FOR TIME=1 TO 10:NEXT TIME
60 NEXT PITCH
70 FOR PITCH=40 TO 60 STEP 2
80 SOUND 0,PITCH,10,8
90 FOR TIME=1 TO 10:NEXT TIME
100 NEXT PITCH
110 NEXT TUNE
```

```
10 REM ** FALLING BOMB **
20 FOR PITCH=30 TO 200
30 SOUND 0,PITCH,10,6
40 NEXT PITCH
50 FOR PITCH=30 TO 200
60 SOUND 0,PITCH,0,6
70 NEXT PITCH
```

```
10 REM ** BIRDS CHIRPING **
20 FOR TUNE=1 TO 10
30 FOR PITCH=1 TO 20
40 SOUND 0,PITCH,10,8
50 NEXT PITCH
60 NEXT TUNE
```

```
10 REM ** WAVES **
20 FOR PITCH=0 TO 10
30 SOUND 0,PITCH,8,4
40 FOR TIME=1 TO 30:NEXT TIME
50 NEXT PITCH
60 FOR PITCH=10 TO 0 STEP -1
70 SOUND 0,PITCH,8,4
80 FOR TIME=1 TO 100:NEXT TIME
90 NEXT PITCH
100 RUN
```

```
10 REM ** CANNON FIRING **
20 FOR TUNE=1 TO 20
30 SOUND 0,80,0,14
40 FOR PITCH=100 TO 1 STEP -1
50 SOUND 0,PITCH/10,0,PITCH/14
60 NEXT PITCH
70 SOUND 0,0,0,0
80 FOR TIME=1 TO RND(1)*200:NEXT TIME
90 NEXT TUNE
```

```
10 FOR VOLUME=11 TO 0 STEP -0.2
20 FOR VOICE=0 TO 3
30 SOUND VOICE,120+VOICE,10,VOLUME
40 NEXT VOICE
50 NEXT VOLUME
```

```
10 FOR TUNE=1 TO 13
20 READ PITCH,TIME1
30 SOUND 0,PITCH,10,8
40 FOR TIME=0 TO 20*TIME1:NEXT TIME
50 SOUND 0,0,0,0
60 NEXT TUNE
70 DATA 60,2,72,2,60,4,60,2,72,2,60,4,
53,2,60,2,64,2,72,2,81,2,72,2,64,2
```

ANTENNA ANALYSIS

By Dennis Sampson

```
1 REM ********************************
2 REM *       ANTENNA ANALYSIS       *
3 REM *      by  Dennis Sampson      *
4 REM * Published by Atari Computer *
5 REM *    Enthusiasts (N.S.W.)     *
6 REM *           June 1985          *
7 REM ********************************
10 DIM N$(10):POKE 709,10:POKE 710,0:?
 CHR$(125)
20 ? "DESIGN OF NORMAL MODE HELLICAL A
NTENNA"
22 ? "  Adapted for ATARI 800XL by Den
nis     Sampson VK2DED from an article
 in"
24 ? "      AMATEUR RADIO MAGAZINE by
                 John F Drew VK5DJ":?
26 ? "'NORMAL MODE' is where the Radia
tion  pattern is at right angles to th
e ";
28 ? "axis (as per a rod or wire aeria
l).":? :?
30 ? "Enter the Frequency in MHz ";:IN
PUT F1
32 W1=299792/F1
34 W2=SQR(W1)
36 W3=W2^5
40 ? "What is the height of winding(mm
) ";:INPUT H1
42 H2=16*H1*H1
50 ? "What is the smallest diameter of
 the  rod(mm) ";:INPUT D1
52 ? "What is the largest diameter for
 the  rod(mm) ";:INPUT D2
54 D3=(D1+D2)/2
56 D4=20*D3*D3*D3
58 A1=(W3/H2)-W2
60 A2=A1/D4
62 N1=INT(A2^0.4*100+0.5)/100
64 P1=INT(10*N1*100+0.5)/100:P3=INT(N1
*3.1416*H1*D3/100*100+0.5)/100
66 P4=INT(N1*H1*100+0.5)/100
68 ? "OUTPUT ON SCREEN OR PRINTER (S o
r P) ";:INPUT N$
70 IF N$(1,1)="S" THEN N$="E:":GOTO 80
72 IF N$(1,1)="P" THEN N$="P:":GOTO 80
74 GOTO 68
80 OPEN #4,8,0,N$
82 ? #4;"Your design criteria are:":?
#4;"Frequency (MHz) ";:? #4;F1
84 ? #4;"Height of winding (mm) = ";:?
 #4;H1
86 ? #4;"Minimum diameter of rod(mm) =
 ";:? #4;D1
88 ? #4;"Maximum diameter of rod(mm) =
 ";:? #4;D2
90 ? #4;"Use wire gauge 14 to 24 SWG"
92 ? #4
94 ? #4;"Turns per mm is ";:? #4;N1
96 ? #4
98 ? #4;"Turns per cm is ";:? #4;P1
100 ? #4
102 ? #4;"Total number of turns is ";:
? #4;P4
104 ? #4
106 ? #4;"Length of wire in METRES is
";:? #4;P3
108 ? #4
110 ? #4;"So there it is...Now go and
build that 'Super Stick'!!"
```

SMARTSHEET

by Ken Shiu

Smartsheet is a spreadsheet calculator which is extremely handy for financial forecasting, budgeting or any calculations that involve many variables. NOTE:- Due to its need for two buffers, Smartsheet will only work on a 32K cassette or 48K disk system as a minimum.

The on-screen worksheet is divided up into cells or grid co-ordinates, arranged 15 columns across (A-O) and 40 rows down. Examples are:- A1, O15, F32, etc. Due to the cell format (total of 600 cells) it is only possible to display a portion of the worksheet on the screen, so in order to view different areas of the worksheet, the screen acts as a scrolling window over the worksheet.

To attempt to explain how to operate a spreadsheet in a few pages is no easy task. Those already familiar with spreadsheets (e.g. VISICALC, SYNCALC), should feel at home since Smartsheet is basically styled after VISICALC. To newcomers, I hope the following will be enough to get you started.

When Smartsheet is run, the screen is divided into two sections. The upper blue screen is the input window, which displays different menus, input prompts, error messages and the current cell co-ordinate. Below, is the grey worksheet screen, the window to the rest of the sheet. The black inverse bar is the cursor and is controlled by the normal cursor control keys. Its initial position is cell, A1.

Smartsheet recognises three cell types: Labels, Values or Formulas. Since Smartsheet only involves itself with number calculations, labels are for the users' benefit, similar to REM's in BASIC. They are usually placed in the column left of a value, to identify it. e.g., SALES, COST, PROFIT, etc. Labels are exactly like the list of items on a shopping list. To enter a label, position the cursor and type in the label, if the label is too long, the cursor will automatically be forwarded to the next column.

Values are numbers, you input for the worksheet calculations to function properly. Values may take any form -positive, negative, decimal, etc. The use of values are similar to the prices next to items on a shopping list. Values are inputed by typing numbers directly into the cell. When the cursor is moved away, the value is moved to the right to allign the decimal places.

By pressing OPTION, formulas can by imputed into the current cell or answers to simple equations can be found. NOTE:- Values are necessary in a worksheet for formula to function at all. Smartsheet gathers its input from cells nominated within a formula and displays the result after all calculations are complete. After pressing OPTION, "Formula" appears on the status line and on the input line you are asked whether the first number in the formula is to be a cell location or a number. The power of formulas in Smartsheet, is its ability to access values from other cells. i.e., formula may calculate a PROFIT figure, and therefore will access the values you inputed for SALES and COST and subtract them.

Next the desired operation has to be inputed- addition, subtraction, multiplication, division or exponent (power of). Smartsheet is limited to one operation per formula. After entering the 2nd number as a cell or number the full equation will be seen on the sheet. If no cells have been accessed, the formulas will remain until they are calculated after pressing START.

A subset of the formula, is the SUM function, which is accessed by typing a colon (:). The SUM function allows you to total values between one cell and another in a particular row or column. After typing a colon, the input line asks "FROM CELL:?", here you should enter the cell where the totalling will begin. e.g. A1. Your input will be registeredin the brackets in the status line. Input the cell, where the totalling will end, when "TO CELL:?" appears. e.g. A9. The input line will be cleared, and when you move the cursor off the formula cell, your From and To cells will be shown. e.g. :A1:A9

Once you have finished structuring your worksheet, complete with labels, values and formulas, press START to calculate the worksheet. The message 'Calculating...' will appear while Smartsheet is computing answers. Calculating time depends on the number of formulas within the worksheet. When Smartsheet has finished, the screen will temporarily clear and the final worksheet will be seen with all formulas replaced with the results.

Smartsheet also has an optional menu for aid while developing a worksheet. The menu is accessed by pressing SELECT. The menu 'G L E S P H' will appear on the status line. Press the corresponding key to obtain these functions.

G - Global Format: Selects how values are to be formatted when inputed. Choose from Dollar, Normal and Integer formatting. Dollar will automatically change your inputed value to dollar and cent format. Normal will leave your value untouched, while Integer will round your input to the next whole number.

L - Load Worksheet: Loads a previously saved worksheet from a disk or cassette. Press D or C to select Disk or Cassette respectively. If using cassette follow the same procedure as loading BASIC programs. If using disk, you may either:-
1. Press the bar to cycle through the Smartsheet workfiles on your disk and press RETURN to load the file displayed in the input window.
2. Input a filename directly on the input line and press RETURN to load it.

S - Save Worksheet: Saves current worksheet in memory to either disk or cassette. Press D or C to select Disk or Cassette to save on respectively. NOTE:- Smartsheet saves the whole sheet, so cassette owners make sure you have about 50 counter spaces on the cassette and be prepared to wait during saving and loading times! Disk owners have two choices (same procedure as Load function):-
1. Press the space bar to cycle through the Smartsheet files on your disk and press RETURN to update or save over the file displayed in the input window.
2. Input a filename (8 letter limit) directly on the input line and press RETURN to save it. NOTE:- Smartsheet uses '.SS' as an extender on its saved worksheets to identify them.

E - Erase Worksheet: Clears the current worksheet from memory. The program will re-ask whether you wish to erase the current worksheet in memory. Type Y to erase, or any other key to return to the worksheet. If you type Y then the screen will temporarily clear and a clean worksheet will appear.

P - Print Worksheet: Prints the current worksheet to a printer. Make sure your printer is ON LINE!! First you will be asked to input the cell at the lower right corner of your worksheet (in order to define the bottom and rightmost column). After entering the cell co-ordinate, you may imbed printer control codes at the beginning of each row in the worksheet. Type Y to imbed control codes e.g. double width for headings. If you elect not to use printer codes, press RETURN to begin printing. If you type Y for printer codes input the row number to imbed the code. NOTE:- The program will send control code before printing the row. Next, type the code in and follow the same procedure to input more codes. When you have finished press RETURN to print the worksheet.

H - Home Cursor: Returns the cursor to cell A1. When you are moving around the far extremes of the worksheet it is handy to use this function instead of repeatedly using the cursor keys.

? - Help Screen: Calling up this screen lists all the main keys and functions of Smartsheet.

That about wraps up the features of Smartsheet. If you prefer to have the cursor move without having to use the CONTROL and arrow keys simultaneously, just change the value equal to K in the lines 65,70,75 and 80 to 61,45,43 and 42 respectively.

EXAMPLE WORKSHEET

An actual example would be better explain the basics behind a worksheet or 'template', ala the two sample screens. Screen 1 shows a template in its raw state with all formulas being uncalculated. All headings and item names are example of labels. NOTE:- Any character including numbers may be made into a label by typing an apostrophe before entering the label.e.g. the line of minus signs beneath the heading.

The price of the items are all values and have been Dollar formatted. The format has been changed to 'Normal' mid-way in creation to prevent quantity values to be in dollar and cent format.

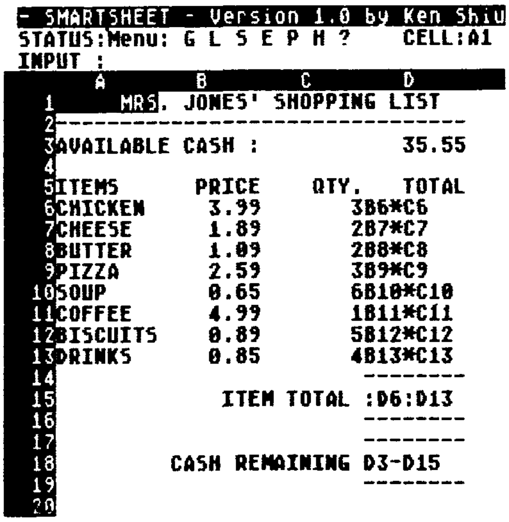
In the D column, formulas are present. In cell D6, the value of cell B6 (price of chicken) will be multiplied by cell C6 (quantity of chicken). The result of this formula will be shown after calculation. The same applies to cell D18, where the item total (D15) is subtracted from the available cash (D3).

Cell D15 (item total) is using the SUM function. Upon calculation, Smartsheet will add all values from cell D6 to D13. NOTE:- Smartsheet calculates all formulas and sums from left to right, top to bottom on the worksheet.

When START is pressed, Smartsheet will pause to calculate and the result will appear as in Screen 2, where all formulas in the D column have been solved and replaced by a number. From here the user may experiment with different cash, price or quantity values to view the final outcome on Mrs Jones' purse. As can be seen in Screen 2, Mrs Jones will have trouble paying the bill with only thirty dollars.

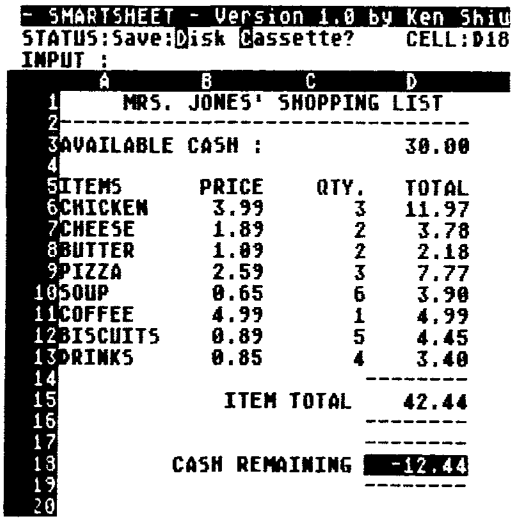
Hopefully, I have made the versatility and applications of spreadsheet more appreciable. The spreadsheet is by far the greatest tool for financial planning. It can definitely save a lot of considerable time and effort. Smartsheet is by no means as powerful as commercial spreadsheets, but it does help to fill the business software gap that Atari owners have been crying out for. I hope that Smartsheet has shown just how much an Atari can help in paying itself off, plus more. So to all you crash-hot (game?) programmers get cracking! INSIDE INFO would like to see you strut your stuff!

SCREEN 1



```
- SMARTSHEET - Version 1.0 by Ken Shiu
STATUS:Menu: G L S E P H ?    CELL:A1
INPUT :
        A         B         C         D
  1          MRS. JONES' SHOPPING LIST
  2--------------------------------------
  3AVAILABLE CASH :                 35.55
  4
  5ITEMS      PRICE    QTY.    TOTAL
  6CHICKEN     3.99         3B6*C6
  7CHEESE      1.89         2B7*C7
  8BUTTER      1.09         2B8*C8
  9PIZZA       2.59         3B9*C9
 10SOUP        0.65         6B10*C10
 11COFFEE      4.99         1B11*C11
 12BISCUITS    0.89         5B12*C12
 13DRINKS      0.85         4B13*C13
 14                          --------
 15          ITEM TOTAL :D6:D13
 16                          --------
 17                          --------
 18        CASH REMAINING D3-D15
 19                          --------
 20
```

SCREEN 2



```
- SMARTSHEET - Version 1.0 by Ken Shiu
STATUS:Save:Disk Cassette?    CELL:D18
INPUT :
        A         B         C         D
  1        MRS. JONES' SHOPPING LIST
  2--------------------------------------
  3AVAILABLE CASH :                 30.00
  4
  5ITEMS      PRICE      QTY.     TOTAL
  6CHICKEN     3.99         3     11.97
  7CHEESE      1.89         2      3.78
  8BUTTER      1.09         2      2.18
  9PIZZA       2.59         3      7.77
 10SOUP        0.65         6      3.90
 11COFFEE      4.99         1      4.99
 12BISCUITS    0.89         5      4.45
 13DRINKS      0.85         4      3.40
 14                               --------
 15          ITEM TOTAL           42.44
 16                               --------
 17                               --------
 18        CASH REMAINING        -12.44
 19                               --------
 20
```

ATASCII Number	ATASCII Character	Necessary Keystrokes			
CHR$(0)	♥	CTRL ,	CHR$(126)	◀	ESC BACK S
CHR$(10)	◣	CTRL J	CHR$(127)	▶	ESC TAB
CHR$(15)	▪	CTRL O	CHR$(138)	◥	INVERSE CTRL J
CHR$(23)	┬	CTRL W	CHR$(141)	■	INVERSE CTRL M
CHR$(24)	┴	CTRL X	CHR$(142)	■	INVERSE CTRL N
CHR$(27)	␛	ESC ESC	CHR$(156)	⍐	ESC SHIFT BACK S
CHR$(28)	↑	ESC CTRL -	CHR$(157)	⍗	ESC SHIFT >
CHR$(30)	←	ESC CTRL +	CHR$(253)	🔔	ESC CTRL 2
CHR$(125)	↰	ESC SHIFT <			

```
1 REM ##################################
2 REM #          SMARTSHEET            #
3 REM #          by  Ken Shiu          #
4 REM # Published by Atari Computer #
5 REM #    Enthusiasts (N.S.W.)     #
6 REM #          June 1985             #
7 REM ##################################
10 DIM OUT$(4800),IN$(4800),S$(600),DL
I$(20),V$(80),H$(15),H(4),CELL$(3),C$(
8),A$(11),FI$(14),DI$(20),PR$(120)
15 GRAPHICS 0:POKE 559,0:OPEN #2,4,0,"
K:":POKE 752,1:POKE 710,127:POKE 709,0
:POKE 712,56:POKE 82,0
20 DL=PEEK(560)+PEEK(561)*256:POKE DL+
7,130:DLI$="H[illegible]e":A=AD
R(DLI$):DIM CL$(8)
25 B=INT(A/256):C=A-B*256:POKE 512,C:P
OKE 513,B:POKE 54286,192:GOSUB 2000
30 ? "↰ - SMARTSHEET - Version 1.0 by
Ken Shiu  STATUS:";:POKE 85,31:? "CELL
:";CELL$
35 ? " INPUT :":? "██████████████████
███████████████████"
40 FOR T=Y TO Y+36 STEP 2:? "█";V$(T,
T+1):NEXT T:? "█";V$(T,T+1);:POSITION
 4,4:? C$
45 FOR T=1 TO 4:POSITION H(T),3:? H$(T
,T):NEXT T:POKE 764,255:POKE 559,34
46 IF FL THEN RETURN
50 TRAP 50:POKE 53774,64:POKE 16,64:IF
 PEEK(764)<>255 THEN 60
54 IF PEEK(53279)=5 THEN POKE 53760,10
0:GOSUB 960:GOSUB 500
56 IF PEEK(53279)=6 THEN POKE 53760,50
:GOSUB 960:GOSUB 1200
57 IF PEEK(53279)=3 THEN POKE 53760,15
0:GOSUB 960:GOSUB 1000
58 GOTO 50
60 GET #2,K
65 IF K=29 THEN GOSUB 200:Y=Y+1:GOSUB
100
70 IF K=28 THEN GOSUB 200:Y=Y-1:GOSUB
100
75 IF K=30 THEN GOSUB 200:X=X-1:GOSUB
150
80 IF K=31 THEN GOSUB 200:X=X+1:GOSUB
150
85 IF (K>47 AND K<58) OR K=45 OR K=43
THEN GOSUB 900
90 IF (K>64 AND K<123) OR K=34 THEN GO
SUB 220
92 IF K=58 THEN GOSUB 800
95 POKE 764,255:POKE 702,64:GOTO 50
100 IF Y>40 THEN Y=40:FOR T=1 TO 10:PO
KE 53761,168:NEXT T:POKE 53761,160:GOS
UB 135:RETURN
105 IF Y<1 THEN Y=1:FOR T=1 TO 10:POKE
 53761,168:NEXT T:POKE 53761,160:GOSUB
 135:RETURN
110 CELL$=CHR$(ASC(H$(X,X))-128):CELL$
(LEN(CELL$)+1)=STR$(Y):POSITION 36,1:?
 ;"   ←←←";CELL$:Y1=Y
115 CP=(X-1)*8+(120*(Y-1))+1:IF Y1>20
AND K=29 THEN Y1=20
116 IF Y1>=YMIN AND Y<=YMAX THEN Y1=20
-(20-(Y-(YMIN-1)))
117 IF Y<YMIN THEN Y1=1
120 IF Y>YMAX THEN YMAX=YMAX+1:YMIN=YM
IN+1:POSITION 0,4:? "⍐":POSITION 0,23:
? "█";V$(Y*2-1,Y*2);OUT$(CP,CP+31);
125 IF Y<YMIN THEN YMIN=YMIN-1:YMAX=YM
AX-1:POSITION 0,4:? "⍗":POSITION 0,4:?
 "█";V$(Y*2-1,Y*2);OUT$(CP,CP+31);
135 POSITION H(X1)-3,Y1+3:FOR T=CP TO
CP+7:C$(T-CP+1,T-CP+1)=CHR$(ASC(OUT$(T
,T))+128):NEXT T
140 ? C$;:POSITION 8,1:T=15*(Y-1)+X:A$
=S$(T,T):IF A$="1" THEN ? "Label:";IN$
(CP,CP+7)
144 IF A$="2" THEN ? "Value:";IN$(CP,C
P+7)
145 IF A$=" " THEN ? "
       "
146 IF A$="3" THEN ? "Formula:";IN$(CP
,CP+7)
148 RETURN
150 IF X>15 THEN X=15:FOR T=1 TO 10:PO
KE 53761,168:NEXT T:POKE 53761,160:GOS
UB 135:RETURN
155 IF X<1 THEN X=1:FOR T=1 TO 10:POKE
```

➡

```
 53761,168:NEXT T:POKE 53761,160:GOSUB
 135:RETURN
160 CELL$=CHR$(ASC(H$(X,X))-128):CELL$
(LEN(CELL$)+1)=STR$(Y):POSITION 36,1:?
;"   ←←←";CELL$:X1=X
165 CP=(X-1)*8+(120*(Y-1))+1
167 IF X1>=XMIN AND X1<=XMAX THEN X1=X
-(XMIN-1):GOTO 190
170 IF X1<XMIN THEN X1=1:XMIN=XMIN-1:X
MAX=XMAX-1
171 IF X1>XMAX THEN X1=4:XMAX=XMAX+1:X
MIN=XMIN+1
172 G=0:FOR T=XMAX-3 TO XMAX:G=G+1:POS
ITION H(G),3:? H$(T,T):NEXT T:G=0
173 G=0:TS=(XMIN-1)*8+(120*(YMIN-1))+1
:FOR T=TS TO TS+2280 STEP 120:G=G+1
174 POSITION 4,G+3:? OUT$(T,T+31);:NEX
T T
190 POSITION H(X1)-3,Y1+3:FOR T=CP TO
CP+7:C$(T-CP+1,T-CP+1)=CHR$(ASC(OUT$(T
,T))+128):NEXT T
192 ? C$;:POSITION 8,1:T=15*(Y-1)+X:A$
=S$(T,T):IF A$="1" THEN ? "Label:";IN$
(CP,CP+7)
194 IF A$="2" THEN ? "Value:";IN$(CP,C
P+7)
196 IF A$=" " THEN ? "
      "
197 IF A$="3" THEN ? "Formula:";IN$(CP
,CP+7)
198 RETURN
200 POSITION H(X1)-3,Y1+3:CP=(X-1)*8+(
120*(Y-1))+1:? OUT$(CP,CP+7);:RETURN
205 POSITION 8,1:? "
     ":RETURN
210 POSITION 8,2:? "
       ":RETURN
220 POSITION 8,1:? "Label":S$(15*(Y-1)
+X,15*(Y-1)+X)="1":T=0:CP=(X-1)*8+(120
*(Y-1))+1
222 C$="████████":OUT$(CP,CP+7)=CL$:IN
$(CP,CP+7)=CL$
223 IF K=34 THEN T=-1:GOTO 235
225 POSITION 8+T,2:? CHR$(K):OUT$(CP+T
,CP+T)=CHR$(K):C$(T+1,T+1)=CHR$(K+128)
226 IN$(CP+T,CP+T)=CHR$(K)
230 POSITION H(X1)-3,Y1+3:? C$;
235 GET #2,K:IF K=155 THEN GOSUB 210:R
ETURN
236 IF K>27 AND K<32 THEN GOSUB 210:PO
P :GOTO 65
238 T=T+1:IF T>7 AND K<>126 THEN 295
240 IF K=126 AND T=8 THEN T=-1:C$="███
█████":OUT$(CP,CP+7)="        ":IN$(CP
,CP+7)="        ":GOSUB 210:GOTO 230
241 IF K=126 THEN T=T-1:IF T<0 THEN T=
0
242 IF K=126 THEN OUT$(CP+T,CP+T)=" ":
C$(T+1,T+1)="█":IN$(CP+T,CP+T)=" ":POS
ITION 8+T,2:? " ":T=T-1:GOTO 230
245 IF (K<32 OR K>122) THEN 235
250 GOTO 225
260 POSITION 8,1:? "Global:Norm Int Do
llar";
265 GET #2,K
270 IF K=27 THEN GOSUB 205:RETURN
275 IF K=78 THEN GL=0:GOSUB 205:RETURN

280 IF K=73 THEN GL=1:GOSUB 205:RETURN
282 IF K=68 THEN GL=2:GOSUB 205:RETURN

285 ? "▣":GOTO 265
290 FOR T=1 TO 200:NEXT T:RETURN
295 GOSUB 210:POP :G=K:K=31:GOSUB 200:
X=X+1:GOSUB 150:K=G:POP :GOTO 65
300 ? "Disk Cassette";
305 GET #2,K
310 IF K=27 THEN GOSUB 205:POP :GOTO 5
0
315 IF K=67 THEN FI$="C:":GOSUB 205:RE
TURN
320 IF K<>68 THEN ? "▣";:GOTO 305
321 POSITION 13,1:? "Input Filename"
322 GET #2,K:IF K=32 THEN 340
325 IF K<65 OR K>90 THEN ? "▣";:GOTO 3
22
327 T=1:GOTO 335
330 GET #2,K:IF K=126 THEN T=1:A$=CL$:
POSITION 8,2:? A$:GOTO 322
331 IF K=155 THEN 338
332 IF (K<48 OR K>90) OR (K>58 AND K<6
5) THEN ? "▣";:GOTO 330
335 POSITION 8+(T-1),2:? CHR$(K):A$(T,
T)=CHR$(K):T=T+1:IF T>8 THEN T=8
337 GOTO 330
338 FOR T=1 TO LEN(A$):IF A$(T,T)=" "
THEN G=LEN(A$):A$(T)="":T=G
339 NEXT T:FI$="D:":A$(LEN(A$)+1)=".SS
":FI$(LEN(FI$)+1)=A$:GOSUB 205:GOSUB 2
10:CLOSE #1:RETURN
340 TRAP 2100:OPEN #1,6,0,"D:*.SS":TRA
P 390
344 INPUT #1;DI$:POKE 54286,192:IF LEN
(DI$)<17 THEN 390
345 POSITION 8,2:? "FILE:";DI$(3,10)
346 GET #2,K:IF K=155 THEN A$=DI$(3,10
):GOTO 338
347 IF K=27 THEN GOSUB 210:POP :RETURN

350 IF K=32 THEN 344
355 GOTO 346
390 POSITION 8,2:? "▣NO MORE SMARTSHEE
T FILES":CLOSE #1:FOR T=1 TO 300:NEXT
T
395 GOSUB 210:GOSUB 205:TRAP 40000:POP
 :RETURN
400 POSITION 8,1:? "Load:";:GOSUB 300
405 TRAP 2100:OPEN #1,4,128,FI$:TRAP 4
20:POSITION 9,1:? "Loading...";A$:T=7
410 A=ADR(IN$):L=4800:GOSUB 570:OUT$=I
N$
415 A=ADR(S$):L=600:GOSUB 570:CLOSE #G
/16:POKE 54286,192
420 POKE 559,0:GOSUB 205:TRAP 40000:GO
SUB 2020:FL=1:GOSUB 30:FL=0:POKE 559,0
425 G=0:TS=(XMIN-1)*8+(120*(YMIN-1))+1
:FOR T=TS TO TS+2280 STEP 120:G=G+1
430 POSITION 4,G+3:? OUT$(T,T+31);:NEX
T T
435 POSITION 4,4:FOR T=TS TO TS+7:C$(T
-TS+1,T-TS+1)=CHR$(ASC(OUT$(T,T))+128)
:NEXT T:? C$;:POKE 559,34:RETURN
450 POSITION 8,1:? "Save:";:GOSUB 300
452 TRAP 2100:OPEN #1,8,128,FI$:POSITI
ON 9,1:? "Writing...";A$:T=11
455 A=ADR(IN$):L=4800:GOSUB 570
458 A=ADR(S$):L=600:GOSUB 570:CLOSE #G
/16:POKE 54286,192
460 GOSUB 205:TRAP 40000:RETURN
500 POSITION 8,1:? "Menu: G L S E P H
? ";
505 GET #2,K
510 IF K=71 THEN 260
512 IF K=76 THEN 400
514 IF K=83 THEN 450
516 IF K=69 THEN 550
518 IF K=80 THEN 600
520 IF K=27 THEN GOSUB 205:RETURN
522 IF K=72 THEN 580
524 IF K=63 OR K=47 THEN 1500
545 ? "▣":GOTO 505
550 POSITION 8,1:? "Erase: Sure? (Y/N)
";
555 GET #2,K
560 IF K=89 THEN POKE 559,0:GOSUB 2000
:POP :GOTO 30
565 GOSUB 205:RETURN
570 G=16:CB=832+G:POKE CB+2,T:HI=INT(A
/256):LO=A-HI*256:POKE CB+4,LO:POKE CB
+5,HI
572 HI=INT(L/256):LO=L-256*HI:POKE CB+
8,LO:POKE CB+9,HI:I=USR(ADR("hhh▣LV▣")
,G)
575 RETURN
580 GOTO 420
600 POSITION 8,1:? "Print: Lower Corne
r?":POSITION 8,2:T=1:A$=""
605 GET #2,K:IF K=27 THEN GOSUB 205:RE
TURN
607 IF K<65 OR K>79 THEN ? "▣";:GOTO 6
05
610 ? CHR$(K);:A$(T,T)=CHR$(K):T=T+1
615 GET #2,K:IF K<48 OR K>57 AND K<>12
6 AND K<>155 THEN ? "▣";:GOTO 615
620 IF T=2 AND K=48 THEN ? "▣";:GOTO 6
15
625 IF K=126 THEN POSITION 8,2:? "   "
:GOTO 600
630 IF K=155 AND T>1 THEN 645
635 IF T=4 THEN ? "▣";:GOTO 615
640 GOTO 610
645 IF VAL(A$(2))>40 THEN ? "▣":GOSUB
210:GOTO 600
650 DI$=A$:GOSUB 210:POSITION 8,1:? "P
rinter Codes? (Y/N)";
655 GET #2,K:IF K=78 THEN 705
660 IF K<>89 THEN ? "▣";:GOTO 650
662 TRAP 662:GOSUB 210:POSITION 8,1:?
"Enter Line Number      ":POSITION 8,2:
INPUT L
665 IF L<1 OR L>40 THEN ? "▣":GOTO 662
667 A$=PR$(L*3-2,L*3):POSITION 14,1:?
"Printer Codes"
670 POSITION 8,2:? "Line ";L;" (ξ";A$(
1,1);"ξ";A$(2,2);"ξ";A$(3);"):   ←←←";
:T=1
675 GET #2,K:IF K=155 THEN 690
680 IF K=126 THEN 670
682 IF T>3 THEN ? "▣":GOTO 675
685 ? "ξ";CHR$(K);:A$(T,T)=CHR$(K):T=T
+1:GOTO 675
690 PR$(L*3-2,L*3)=A$:GOSUB 210:POSITI
ON 14,1:? "More Codes?(Y/N)";
692 GET #2,K:IF K=89 THEN 662
695 IF K<>78 THEN ? "▣";:GOTO 692
700 TRAP 40000
705 G=VAL(DI$(2)):A$=DI$(1,1):FOR T=1
TO 15:IF ASC(H$(T,T))-128=ASC(A$) THEN
 L=T*8:T=15
```

➔

```
710 NEXT T:GOSUB 210:GOSUB 205
720 TRAP 2100:OPEN #1,8,0,"P:":POSITIO
N 8,1:? "Press RETURN to print"
725 GET #2,K:IF K<>155 THEN GOSUB 210:
GOSUB 205:RETURN
730 L1=1:FOR T=1 TO G:A$=PR$(T*3-2,T*3
)
735 FOR Z=1 TO 3:IF A$(Z,Z)=" " THEN N
EXT Z:GOTO 740
736 ? #1;A$(Z,Z);:NEXT Z
740 ? #1;OUT$(L1,L):L=L+120:L1=L1+120:
NEXT T:CLOSE #1
750 GOSUB 205:GOSUB 210:POKE 54286,192
:RETURN
800 S$(15*(Y-1)+X,15*(Y-1)+X)="3":T=0:
CP=(X-1)*8+(120*(Y-1))+1:POSITION 8,1:
? "Sum:(   -   )"
805 POSITION 8,2:? "From Cell:   <<<";
:INPUT A$
810 IF LEN(A$)>3 OR LEN(A$)<2 THEN ? "
■";:GOTO 805
812 IF ASC(A$(1,1))<65 OR ASC(A$(1,1))
>79 THEN ? "↑■";:GOTO 805
814 IF VAL(A$(2))<1 OR VAL(A$(2))>40 T
HEN ? "■";:GOTO 805
818 POSITION 13,1:? A$
820 POSITION 8,2:? "To Cell  :   <<<";
:INPUT DI$
822 IF LEN(DI$)>3 OR LEN(DI$)<2 THEN ?
 "■";:GOTO 820
824 IF ASC(DI$(1,1))<65 OR ASC(DI$(1,1
))>79 THEN ? "■";:GOTO 820
826 IF VAL(DI$(2))<1 OR VAL(DI$(2))>40
 THEN ? "■";:GOTO 820
827 IF A$(1,1)<>DI$(1,1) AND A$(2)<>DI
$(2) THEN POSITION 8,2:? "■Cells must
be linear!":GOSUB 290:GOSUB 210:GOTO 8
00
828 POSITION 17,1:? DI$
830 FI$=":":A$(LEN(A$)+1)=":":A$(LEN(A
$)+1)=DI$:FI$(LEN(FI$)+1)=A$
835 IF LEN(FI$)<8 THEN FI$(LEN(FI$)+1)
="  "
840 OUT$(CP,CP+7)=FI$:IN$(CP,CP+7)=FI$
850 GOSUB 210:GOSUB 205:RETURN
900 S$(15*(Y-1)+X,15*(Y-1)+X)="2":T=0:
CP=(X-1)*8+(120*(Y-1))+1:POSITION 8,1:
? "Value:";IN$(CP,CP+7):C$="████████"
902 G=0:GOSUB 210:IN$(CP,CP+7)=CL$:OUT
$(CP,CP+7)=CL$:IF K=126 THEN T=-1:GOTO
 910
905 POSITION 8+T,2:? CHR$(K):C$(T+1,T+
1)=CHR$(K+128):IN$(CP+T,CP+T)=CHR$(K):
OUT$(CP+T,CP+T)=CHR$(K)
910 POSITION H(X)-3,Y+3:? C$;
920 GET #2,K:T=T+1:IF K=155 OR (K>27 A
ND K<32) THEN GOSUB 210:GOTO 940
922 IF T>7 THEN ? "■";:GOTO 920
926 IF K=126 THEN 900
928 IF (K>47 AND K<58) THEN 905
930 IF K=46 AND G=1 THEN 935
932 IF K=46 THEN G=1:GOTO 905
935 ? "■";:GOTO 920
940 IF K=155 AND  NOT T THEN RETURN
941 IF  NOT T THEN POP :GOTO 65
942 TRAP 955:A=VAL(IN$(CP,CP+7)):IF GL
=0 THEN A$=STR$(A):GOTO 948
943 IF GL=2 THEN 945
944 A=INT(A+0.5):A$=STR$(A):L=8-LEN(A$
):GOTO 948
945 IF A=INT(A) THEN A$=STR$(A):A$(LEN
(A$)+1)=".00":GOTO 948
946 IF INT(A*10)=A*10 THEN A$=STR$(A):
A$(LEN(A$)+1)="0":GOTO 948
947 A=INT(100*A+0.5)/100:A$=STR$(A)
948 L=8-LEN(A$):OUT$(CP,CP+L)=CL$:OUT$
(CP+L,CP+7)=A$:IN$(CP,CP+7)=OUT$(CP,CP
+7):TRAP 40000
950 IF K=155 THEN RETURN
952 POP :GOTO 65
955 ? "■";:K=126:GOTO 900
960 FOR L=1 TO 10:POKE 53761,175:NEXT
L:POKE 53761,160:POKE 53760,200:RETURN

1000 T=0:CP=(X-1)*8+(120*(Y-1))+1:POSI
TION 8,1:? "Formula:";IN$(CP,CP+7):G=0
:O=G:Z=1
1004 POSITION 8,2:? "1st No.:Cell or N
umber";:GET #2,K:IF K=27 THEN GOSUB 21
0:GOSUB 205:RETURN
1005 IN$(CP,CP+7)=CL$:IF K=67 THEN G=1
:GOSUB 1020:GOTO 1030
1006 IF K<>78 THEN ? "■";:GOTO 1004
1007 GOSUB 1008:L=A:GOTO 1030
1008 TRAP 1008:GOSUB 210:POSITION 8,2:
? "Enter No.:";:INPUT A$:IF LEN(A$)>4
THEN ? "↑■";:GOTO 1008
1009 POSITION 16+T,1:? A$:K=LEN(A$)-1:
A=VAL(A$):IF G OR Z THEN IN$(CP+T,CP+T
+K)=A$:T=T+K+1:RETURN
1010 IF O=1 THEN A=A+L
1011 IF O=2 THEN A=L-A
1012 IF O=3 THEN A=A*L
1013 IF O=4 THEN A=L/A
1014 IF O=5 THEN A=INT((L^A)+0.5)
1015 IN$(CP,CP+7)=CL$:IN$(CP,CP+7)=STR
$(A):K=155:GOSUB 942:GOSUB 210:S$(15*(
Y-1)+X,15*(Y-1)+X)="2":RETURN
1020 GOSUB 210:POSITION 8,2:? "Enter C
ell:";:INPUT A$:IF LEN(A$)<2 OR LEN(A$
)>3 THEN 1025
1021 IF ASC(A$(1,1))<65 OR ASC(A$(1,1)
)>80 THEN 1025
1022 IF VAL(A$(2))<1 OR VAL(A$(2))>40
THEN 1025
1024 GOTO 1026
1025 ? "↑■";:GOTO 1020
1026 K=LEN(A$)-1:IN$(CP+T,CP+T+K)=A$:P
OSITION 16+T,1:? A$:T=T+K+1:RETURN
1030 S$(15*(Y-1)+X,15*(Y-1)+X)="3":GOS
UB 210:POSITION 8,2:? "Operation:+ - *
 / ^";
1031 GET #2,K:IF K=43 THEN O=1:A$="+":
GOTO 1040
1032 IF K=45 THEN O=2:A$="-":GOTO 1040
1033 IF K=42 THEN O=3:A$="*":GOTO 1040
1034 IF K=47 THEN O=4:A$="/":GOTO 1040
1035 IF K=94 THEN O=5:A$="^":GOTO 1040
1036 ? "■";:GOTO 1031
1040 Z=0:IN$(CP+T,CP+T)=A$:POSITION 16
+T,1:? A$:T=T+1:POSITION 8,2:? "2nd No
.:Cell or Number";
1042 GET #2,K:IF K=67 THEN GOSUB 1020:
GOTO 1050
1044 IF K=78 THEN GOSUB 1008:GOTO 1050
1045 ? "■";:GOTO 1042
1050 GOSUB 210:OUT$(CP,CP+7)=IN$(CP,CP
+7):RETURN
1200 POSITION 8,1:? "   Calculating...
";
1202 FOR T=1 TO 600:IF S$(T,T)="3" THE
N GOSUB 1210
1205 NEXT T:GOTO 420
1210 CP=(T-1)*8+1:A$=CL$:IF IN$(CP,CP)
=":" THEN 1400
1212 FOR A=CP TO CP+3:L=ASC(IN$(A,A)):
IF L=42 OR L=43 OR L=45 OR L=47 OR L=9
4 THEN Z=A-CP+1:A=CP+3:O=L:GOTO 1216
1214 A$(A-CP+1,A-CP+1)=CHR$(L):NEXT A
1216 TRAP 40000:IF ASC(A$(1,1))>64 AND
 ASC(A$(1,1))<80 THEN 1220
1218 A=VAL(A$):GOTO 1230
1220 TRAP 1225:F=ASC(A$(1,1))-64:G=VAL
(A$(2)):H=(F-1)*8+(120*(G-1))+1:A=VAL(
IN$(H,H+7)):GOTO 1230
1225 TRAP 1226:A=VAL(OUT$(H,H+7)):GOTO
 1230
1226 A=0
1230 A$=CL$:A$=IN$(CP+Z,CP+7):IF ASC(A
$(1,1))>64 AND ASC(A$(1,1))<80 THEN 12
34
1232 L=VAL(A$):GOTO 1240
1234 TRAP 1235:F=ASC(A$(1,1))-64:G=VAL
(A$(2)):H=(F-1)*8+(120*(G-1))+1:L=VAL(
IN$(H,H+7)):GOTO 1240
1235 TRAP 1236:L=VAL(OUT$(H,H+7)):GOTO
 1240
1236 L=0
1240 TRAP 40000:IF O=42 THEN A=A*L
1242 IF O=43 THEN A=A+L
1244 IF O=45 THEN A=A-L
1246 IF O=47 THEN A=A/L
1247 IF O=94 THEN A=INT((A^L)+0.5)
1248 IF GL=0 THEN A$=STR$(A):GOTO 1270
1250 IF GL=2 THEN 1260
1252 A=INT(A+0.5):A$=STR$(A):L=8-LEN(A
$):GOTO 1270
1260 IF A=INT(A) THEN A$=STR$(A):A$(LE
N(A$)+1)=".00":GOTO 1270
1262 IF INT(A*10)=A*10 THEN A$=STR$(A)
:A$(LEN(A$)+1)="0":GOTO 1270
1264 A=INT(100*A+0.5)/100:A$=STR$(A)
1270 L=8-LEN(A$):OUT$(CP,CP+L)=CL$:OUT
$(CP+L,CP+7)=A$:RETURN
1400 SUM=0:A$=IN$(CP+1,CP+3):H=3:IF A$
(3,3)=":" THEN A$=IN$(CP+1,CP+2):H=2
1402 A=ASC(A$(1,1))-64:L=VAL(A$(2))
1404 A$=IN$(CP+2+H,CP+7):F=ASC(A$(1,1)
)-64:G=VAL(A$(2))
1405 IF A=F THEN O=A-1:FOR Z=L TO G:TR
AP 1416:H=O*8+(120*(Z-1))+1:V=VAL(IN$(
H,H+7)):GOTO 1420
1410 IF L=G THEN O=L:FOR Z=A-1 TO F:TR
AP 1416:H=Z*8+(120*(O-1))+1:V=VAL(IN$(
H,H+7)):GOTO 1420
1416 TRAP 1417:V=VAL(OUT$(H,H+7)):GOTO
 1420
1417 V=0
1420 SUM=SUM+V:NEXT Z:A=SUM:GOSUB 1248
:RETURN
1500 ? "K":? ")- SMARTSHEET HELP SCREE
N -":? :?
1505 ? "  NORMAL CURSOR ARROW KEYS CON
TROL THE":? "  CURSOR (LONG BLACK BAR)
."
1510 ? "  A...Z,";CHR$(34);"    Marks
cell as a label":? "  0...9,+,-  Marks
 cell as a value"
1515 ? "  :             Sum Formula. Tota
ls values":? "              in a column
```

➡

```
 or row."
1520 ? "  OPTION     Marks cell as a f
ormula":? "  SELECT     Main Menu. Cho
ose from:-"
1525 ? "                G=Global Format":
? "            L=Load Worksheet":? "
          S=Save Worksheet"
1530 ? "                P=Print Worksheet
":? "            E=Erase Worksheet"
1532 ? "                H=Home Cursor.(Ce
ll A1)":? "               ?=Help Screen.
"
1535 ? "  START     Calculate Workshe
et":? :? :? ")- PRESS ANY KEY TO RETUR
N -";:GET #2,K:GOTO 420
2000 V$=" 1 2 3 4 5 6 7 8 9101112131415
161718192021222324252627282930313233343
5363738394 0"
2005 N$="ABCDEFGHIJKLMNO":N(1)=7:FOR T
=2 TO 4:N(T)=N(T-1)+8:NEXT T:POKE 5376
0,200:GL=0
2010 OUT$(1)=" ":OUT$(4800)=" ":OUT$(2
)=OUT$:IN$=OUT$:S$(1)=" ":S$(600)=" ":
S$(2)=S$
2015 PR$(1)=" ":PR$(120)=" ":PR$(2)=PR
$
2020 X=1:Y=1:C$="          ":XMIN=X:YMIN
=X:Y1=X:X1=X:XMAX=4:YMAX=20:CELL$="A1"
:FL=0:CL$="          "
2050 RETURN
2100 CLOSE #1:RESTORE 2120:POKE 54286,
192
2105 READ E,DI$:IF E=PEEK(195) THEN PO
SITION 8,1:? "Error:";DI$:TRAP 40000:
GOSUB 210:GOTO 50
2110 IF E=-1 THEN POSITION 8,1:? "Err
or:No.";PEEK(195);"          ":TRAP 4000
0:GOSUB 210:GOTO 50
2115 GOTO 2105
2120 DATA 138,Device Not On,139,Device
 NAK,140,Serial Frame,142,Serial Bus,1
43,Checksum No.143
2122 DATA 144,Write Protected,162,Disk
 Full,163,Bad DOS,167,File Locked,170,
No such file,-1,a
```

NO TURTLE GRAPHICS... BUT IT'S STILL LOGO!

by Arthur Banks

Most of my programming is done in Logo. As a result I have felt reluctant to submit any programs to "Inside Info". No-one else I know is interested in Logo. Everyone seems to think that Logo is just turtle graphics and just kids' stuff.

Now both of these ideas are quite wrong. Logo is far more than turtle graphics and, although it can be introduced to children, it is also a very powerful language in its own right.

As an example of a non-graphics application of Logo, here is a set of procedures to analyse the frequency with which each letter of the alphabet appears in a piece of text.

A program in Logo consists of a controlling procedure which manages the program as a whole by calling a number of sub-procedures at the appropriate times.

The controlling procedure in this program is called ANALYSE.LETTERS. Here it is-

```
TO ANALYSE.LETTERS
MAKE "LETTER.LIST []
GET.TEXT
COUNT.LETTERS :TEXT
REPORT :LETTER.LIST
END
```

Each procedure must have a name. This one, ANALYSE.LETTERS, first establishes a variable called LETTER.LIST which, to begin with, has no contents. (It will ultimately contain a list of the letters in the text to be analysed.) ANALYSE.LETTERS next calls in turn three sub-procedures named GET.TEXT, COUNT.LETTERS and REPORT.

GET.TEXT does what its name implies. It accepts text from the keyboard.

```
TO GET.TEXT
PR [TYPE IN UP TO THREE LINES OF TEXT]
PR [FOLLOW EACH ENTRY WITH RETURN]
PR [WHEN FINISHED ENTERING ALTOGETHER TYPE AN ASTERISK]
PR [FOLLOWED BY A RETURN]
MAKE "TEXT []
ACCEPT.TEXT
END
```

GET.TEXT itself contains a subprocedure-

```
TO ACCEPT.TEXT
MAKE "ENTRY RL
IF :ENTRY = [*] [STOP]
MAKE "TEXT SE :TEXT :ENTRY
ACCEPT.TEXT
END
```

➽

MAKE "ENTRY RL sets up a variable called entry and accepts input from the keyboard through the command RL (READLIST). If the entry is an asterisk this subprocedure stops and control goes back to the procedure which called it. If the entry is not an asterisk the contents of entry is added to what is already contained in the variable called text. ACCEPT.TEXT then calls itself so that the process is continued.

Once the text has been entered ANALYSE.LETTERS then calls the next procedure-

```
TO COUNT.LETTERS :TEXT
IF EMPTYP :TEXT [STOP]
GET.NEXT.WORD
GET.NEXT.LETTER :NEXT.WORD
COUNT.LETTERS BF :TEXT
END
```

The first line stops this procedure if and when the text variable is empty. If it is not empty then it calls GET.NEXT.WORD. From that word it gets the next letter using GET.NEXT.LETTER and then calls itself to repeat the process.
Here is GET.NEXT.WORD.

```
TO GET.NEXT.WORD
MAKE "NEXT.WORD FIRST :TEXT
END
```

This isn't very long is it? All it does is take the first word of what remains of the text.

```
TO GET.NEXT.LETTER :NEXT.WD
IF EMPTYP :NEXT.WD [STOP]
MAKE "NEXT FIRST :NEXT.WD
IF MEMBERP :NEXT :LETTER.LIST [MAKE :NEXT (THING :NEXT) + 1] [MAKE "LETTER.LIST LPUT :NEXT
:LETTER.LIST MAKE :NEXT 1]
GET.NEXT.LETTER BF :NEXT.WD
END
```

If the word being examined at the moment is empty then this procedure ends and the program returns to GET.NEXT.WORD. If any letters remain in the current word then the procedure takes the first of the remaining letters and stores it in a variable called NEXT. If that letter is already contained in the list of letters found in the text to date then the procedure adds one to the count for that letter. If this is the first occurrence of that letter then the letter is added to the list of letters found in the text and the count for that letter is set to one.

The final procedure called in this program is-

```
TO REPORT :LETTER.LIST
IF EMPTYP :LETTER.LIST [STOP]
PR SE FIRST :LETTER.LIST THING FIRST :LETTER.LIST
REPORT BF :LETTER.LIST
END
```

What this does, for each letter found in the text, is print out the letter and the number of times that letter was found

I realise that this program probably looks fairly complicated but it really isn't particularly difficult and it does show that Logo can handle text as well as the graphics for which it is better known.

SIMPLE DOLLAR FORMATTING

By Peter Egan

```
10 DIM D$(20)
20 FOR D=1 TO 10 STEP 0.05:REM DEMO LOOP
30 IF ABS(D)<1.0E-03 THEN D$="0.00":GOTO 50
40 D$=STR$(D+5.0E-03):D$=D$(1,LEN(STR$(INT(D)))+3)
50 PRINT "$";D$
60 NEXT D
```

adventure!

by Garry Francis

NEW ADVENTURE DISK!

The big news this month is the addition of a disk of Adventures to the Software Exchange. See SOFTWARE EXCHANGE NEWS elsewhere in this issue for details.

APX ADVENTURES DIRT CHEAP

It is not generally known, but the Atari Program Exchange sells a small range of text-based Adventures. These are fairly easy Adventures, which makes them ideal for beginners. They are written in BASIC, but because of the disk interaction, they are available only on disk. After the demise of Futuretronics, the receivers were selling off all stocks of Atari and APX programs at ridiculously low prices. I recently found some of the APX Adventures being sold at Computer Wave, 325 George Street, Sydney for $15, marked down to $10, but I offered them $5 and they accepted! At that price, you'd be crazy not to buy everything you can get your hands on! The Adventures to look out for are Wizard's Gold, Sleazy Adventure, Alien Egg, Chinese Puzzle, Castle, Sultan's Palace and Wizard's Revenge.

NEW ADVENTURE FROM ATARI CORP.

I often refer to the English users' magazine called Page'6. The last issue (issue 14) had a couple of advertisements for a new Adventure from Atari Corporation. It's called Pay-Off and is available only on disk at 9.95 pounds in the U.K.

NEW RELEASES FROM SYNAPSE

Synapse have released a series of sophisticated all-text Adventures which they refer to as Electronic Novels. They have a full sentence parser which is apparently as sophisticated as Infocom's. In fact, one review said it was "lightyears ahead of its competition".

The first titles in the series are Mindwheel (a journey into the minds of four deceased people to find clues to the Wheel of Wisdom) and Essex (an intergalactic search and rescue mission). Future titles include Brimstone (a medieval adventure story), Breakers (a science fiction fantasy on the planet Borg) and Ronin (a samurai epic). The Atari versions are priced at US$39.95, which includes a hardcover book which you need to read before playing the game!

DO-IT-YOURSELF ADVENTURES

Ever wanted to write your own Adventure, but didn't have the programming expertise? Have no fear. There are now a number of Adventure writing programs on the market. I haven't seen any of these "in the flesh" yet, but they have certainly generated a lot of interest from other people. First in line is Adventure Master from CBS Software. One of our members has bought it and he was raving over it at the Australian Personal Computer Show.

Second in line is Adventure Writer from Code Writer Corporation. The only review I've read so far was in Mideastern Software's newsletter and they were wrapped in it. According to the January 1985 issue of Micro Adventurer, Adventure Writer is a disk-based version of the English program called The Quill (which was originally written for the Spectrum), but I'm not sure how true this is. The Quill has had some good reviews, but the commercial Adventures produced with it have generally been poor. Let's hope Atari users are more imaginative.

Finally, Electronic Arts has announced Adventure Construction Set. If it's as good as other Electronic Arts products, then this could be the pick of the crop, but who knows? If you write any Adventures using any of these products, please send them to me for the Software Exchange.

ORIGINAL ADVENTURE FOR THE ATARI

Q: Can I get a copy of "Adventure" for the Atari? I've played it on the University of New England's DEC 20 and become addicted. - Danny McClatchey

A: You bet your brass lantern you can! I know of two versions for the Atari. The first is called Original Adventure and is written in BASIC by Robert A. Howell. It is available direct from the author on a 32k cassette or 40k disk for US$20.00. The price includes return airmail postage and an 8 page manual with loading and playing instructions, brief history and 86 coded hints. Send to Robert A. Howell, 20 Richman Road, Hudson, NH 03051, U.S.A. and tell him you read about it in INSIDE INFO! (A full review, as well as a history of Adventure and coded hints are given in my Adventure column in Page 6 Issue 9. There should be a copy in the library, otherwise write to Page 6, P.O. Box 54, Stafford, ST16 1DR, U.K. for subscription details.)

➞

The second version is called Colossal Adventure and extends the original mainframe version by adding a new end-game with 70 additional rooms! It costs 9.95 pounds and is available only on cassette from Level 9 Computing, 229 Hughenden Road, High Wycombe, Bucks. HP13 5PG, U.K.

HINTS

Q: I've got 12 of the 13 treasures in Adventureland. Is the last treasure in the lava? If so, then how do I get it? - Matthew Johnson

A: I'll be dammed if you think I'm going to tell you where the last treasure is!

* * * * *

Q: I've got two questions about The Curse of Crowley Manor. How do you open the rosewood chest in the parlour and how do you get through the plywood wall in the small storage room? - Gary Owen

A: Remember the growth in the kitchen? Take it to the dining room and let it have its fill! This should answer both your questions.

* * * * *

Q: What's the third thing you need to do to get through the Entrance to Hades in Zork I? - Matthew Johnson

A: Sir Richard Steele once wrote that "reading is to the mind what exorcise is to the body" ... or something like that.

* * * * *

Q: How do you find your way around the oddly-angled rooms in Zork II? Has it got anything to do with baseball? - Daryl Barlow

A: Yes, it does have something to do with baseball. First of all, find the home plate (now what would you expect to find at the home plate?), then walk in the directions that you'd expect to follow if actually playing a game of baseball. If it's any consolation, the Zork II Invisiclues tell us that the authors designed this maze to confound maze mappers. (It certainly confounded me!) They admit it's a difficult problem and apologise to non-American Zorkers who wouldn't be familiar with all the baseball oriented hints.

* * * * *

Q: I'm having trouble getting started in Starcross. The computer tells me to refer to the mass detector output for the coordinates of the unidentified mass, but I can't find it anywhere. What's going on? - Les Pobje

A: The mass detector output is the map that came with the documentation. (It's labelled in the upper right hand corner.)

* * * * *

Q: I've made my way through the coalmine in Sorcerer, but now that I've reached the top of the chute, I can't think what to do next. Any hints? - Peter Megarrity

A: How did you ever get roped into this?

* * * * *

Q: How do you prove your intelligence to the Sirius Cybernetics door in The Hitchhiker's Guide to the Galaxy? - Bruce Morgan

A: Two other people asked me the same question, but I was unable to help them because I haven't played Hitchhiker's Guide yet. However, I recently received a letter from John Sweeney of Nottingham in the U.K. as a result of my Adventure column in Page 6. John had this to say: "My major problem was how to persuade a door that I was intelligent! I got stuck. During the next three weeks, I solved Suspended and Suspect while continually failing to make any progress on The Hitchhiker's Guide to the Galaxy. Then one day, chatting over lunch, INSPIRATION! If I tried 'that' which was so completely illogical, but logically so (like learning to fly by throwing yourself at the ground and missing!) and assume that all those clues that it had been apparently giving me were actually red herrings and that the jokes were actually clues, then could 'that' actually work? Yes, it did! I am now standing on Magrathea waiting for them to release part II. Infocom prove that they are best yet again!"

* * * * *

Q: When I'm on the Heart of Gold in The Hitchhiker's Guide to the Galaxy, I keep getting transported to different locations. What's going on? - Paul Benjamin

A: I don't know. As I said above, I haven't played Hitchhiker's Guide yet. If anyone can help, please ring Paul on (02) 32-0751.

Have YOU submitted an article yet???

ASSEMBLY LANGUAGE

by Roberto Romano

This is the first of a series of articles on assembly language. This article will focus on macro assemblers (particularly MAC/65 from OSS Software) although general purpose assemblers can also be applied. Those of you who are programming in assembly language without a macro assembler are just wasting a lot of time. With a macro assembler, you can define your own little routines to do anything you want. Once you have written a bit of code which you think might be of some use in the future, you can "crystallise" it in a macro and call that same bit of code over and over without having to re-write it or even re-type it.

How do we write macros ?. I hope to give you many ideas in these articles on how to write and use macros. So go out and buy the MAC/65 and we can get started...

First, I have a couple of useful maths macros.: a fast, efficient multiplication macro and a square root macro. Yes, square root ! Not floating point though. This macro returns the square root of any 8-bit or 16-bit number to the nearest integer and is useful in plotting where square roots are involved in the calculations.

```
10 **************************
20 ** MATHS MACRO LIBRARY **
30 **************************
40 ;
50 ;
60 ** 8-BIT MULTIPLICATION MACRO:
70 ;
80 ; call is : MULT num1,num2,lsb(resu
lt)
90 ; num1 and num2 are literals if the
y are less than 256, else they are mem
ory locations.
0100 ;
0110     .MACRO MULT
0120 ;
0130     JMP @MSTART
0140 @TEMP .BYTE 0
0150 ;
0160 @MSTART .IF %1<256
0170       LDA #%1
0180       .ELSE
0190       LDA %1
0200       .ENDIF
0210       .IF %2<256
0220       LDX #%2
0230       .ELSE
0240       LDX %2
0250       .ENDIF
0260     STA @TEMP
0270     STX %3
0280     LDA #0
0290     STA %3+1
0300     LDY #9
0310     CLC
0320 MULT1 ROR %3+1
0330     ROR %3
0340     DEY
0350     BEQ RET
0360     BCC MULT1
0370     LDA %3+1
0380     CLC
0390     ADC @TEMP
0400     STA %3+1
0410     CLC
0420     BCC MULT1
0430 RET
0440     .ENDM
0450 ;
0460 ;------------------------
0470 ;
0480 ** SQUARE ROOT MACRO **
0490 ;
0500 ; call is SQR num,lsb of loc.(res
ult)
0510 ; num is a memory loc. containing
 the number in lsb/msb form.
0520 ;
0530     .MACRO SQR
0540 ;
0550     JMP @PROCEED
0560 @TMP1 .WORD 0
0570 @TMP2 .WORD 0
0580 @TMP3 .WORD 0
0590 ;
0600 ;
0610 @PROCEED
0620     CLC
0630     LDA @TMP3
0640     ADC @TMP1
0650     STA @TMP3
0660     LDA @TMP3+1
0670     ADC @TMP1+1
0680     STA @TMP3+1
0690     CMP %1+1
0700     BCC CONT
0710     CLC
0720     LDA @TMP3
0730     CMP %1
0740     BCS END
0750 CONT INC @TMP2
0760     BCC CONT2
0770     INC @TMP2+1
0780 CONT2 CLC
0790     LDA @TMP1
0800     ADC #2
0810     STA @TMP1
0820     BCC OK
0830     INC @TMP1+1
0840 OK  JMP @PROCEED
0850 END LDA @TMP2
0860     STA %2
0870     LDA @TMP2+1
0880     STA %2+1
0890     .ENDM
0900 ;
0910 ;============================
0920 ;
0930 ;      *** EXAMPLES ***
0940 ;
0950 NUM1 =  $6000
0960 NUM2 =  $6100
0970 RESULT = $6200
0980 SQRES = $6400
0990     *=  $6300
1000 SQNUM .WORD 0
1010 ;
1020     *=  $4000
1030     .OPT OBJ
1040 ;
1050 ; mult. 30X20, result in $6200
1060 ;
1070     LDA #30
1080     STA NUM1
1090     LDA #20
1100     STA NUM2
1110      MULT  NUM1,NUM2,RESULT
1120 ;
1130 ; sqrt. of 900, result in $6400
1140 ;
1150     LDA # <900
1160     STA SQNUM
1170     LDA # >900
1180     STA SQNUM+1
1190      SQR  SQNUM,SQRES
1200     BRK
```

➸

Those of you not familiar with MAC/65 will notice two unusual symbols in Listings 1 and 2. The first one, % denotes a parameter passed to a macro. That's either a literal value or a memory location. The number of parameters is denoted by %0. The other symbol is @. This denotes a label within a macro and it is good practice to name macro labels in such a way as to easily distinguish them from labels in the rest of the program.

A good exercise for those who don't know MAC/65 is to try and remove the macro-specific code in the MULT macro of Listing 1 and extract the raw 6502 code. Those who succeed will get a free multi-coloured player-missile !!

As this is a column intended for advanced users I will move on rather quickly...

A fairly recent addition to OSS's superb line-up of software is the MAC/65 TOOLKIT. This is a collection of macros that complement the cartridge and allow the skilful programmer to harness the full power of the machine without too much effort. It is disk based and comes with a 29-page booklet which details all the macros and includes examples and hints on how to use them. All the usual I/o macros are present along with page-clear, page-move, block-clear and block-move macros. For example, PGMOVE allows to move a page of memory, e.g. to move page 6 to page 150 you simply say:

```
        PGMOVE 6,150
```

Naturally, Player/missiles are supported. There are 12 P/M macros, among which is PMMOVE, which allows you to move a player to any screen position. For example, PMMOVE 1,150,80 will move player 1 to screen position 150,80 (similar to a BASIC XL command).

For the icing on the cake, OSS have added two macros that allow fine scrolling. SCRDIM sets the fine scrolling parameters and SETXY does a coarse scroll. The scrolling routine itself is executed as a deferred vertical blank interrupt routine.

```
10 ;*************************
20 ;** A GRAPHICS macro  **
30 ;*************************
40 ;
50         *=  $4000
60         .OPT NO LIST
70 ;
80         JMP INCIP
90 ;
0100       .INCLUDE #D4:KERNEL.M65
0110 ;
0120 XC  .WORD 0      ; x-coord of star
t
0130 YC  .WORD 0      ; y-coord of star
t
0140 LEN .WORD 0      ; length of side
0150 ;
0160 ;<><><><><><><><><>
0170 ;
0180       .MACRO FRAME
0190 ;
0200       JMP @BSTART
0210 @T1 .WORD 0
0220 @T2 .WORD 0
0230 @BSTART
0240        CALC  %3
0250        PLUS  %1
0260        STORE  @T1
0270        CALC  %3
0280        PLUS  %2
0290        STORE  @T2
0300        PLOT  %1,%2
0310        DRAWTO  @T1,%2
0320        DRAWTO  @T1,@T2
0330        DRAWTO  %1,@T2
0340        DRAWTO  %1,%2
0350       .ENDM
0360 ;
0370 ;<><><><><><><><><>
0380 ;
0390 INCIP  POKE  YC,40 ;always <192
0400        VPOKE  XC,160 ; maybe >256
0410        GR  10      ; like BASIC
0420        SETCOLOR  2,0,0 :  "    "
0430        COLOR  1   ;  "    "
0440        POKE  87,8 : a bit of colour
!
0450        POKE  706,56
0460        POKE  711,168
0470        POKE  712,152
0480 ONCE  FRAME  XC,YC,LEN
0490        CALC  XC
0500        PLUS  10
0510        STORE  XC
0520        CALC  YC
0530        PLUS  5
0540        STORE  YC
0550        CALC  LEN
0560        PLUS  3
0570        STORE  LEN
0580        IFGT  XC,240,AGAIN ;test ">"
0590       JMP ONCE     ; loop
0600 ;
0610 AGAIN  CALC  XC
0620        MINUS  120
0630        STORE  XC
0640        CALC  YC
0650        MINUS  50
0660        STORE  YC
0670 ;
0680 MORE  FRAME  XC,YC,LEN
0690        CALC  XC
0700        MINUS  10
0710        STORE  XC
0720        CALC  YC
0730        PLUS  5
0740        STORE  YC
0750        CALC  LEN
0760        PLUS  5
0770        STORE  LEN
0780        IFLT  XC,10,END
0790        IFGT  YC,190,END
0800       JMP MORE
0810 END JMP END
```

Listing 2 shows how to use some of the TOOLKIT macros to define a FRAME macro, a frame being an empty square box drawn clockwise from the top left corner. In this listing you will notice some familiar commands like COLOR,SETCOLOR, GR (graphics) and POKE (lines 410 to 470). Other macros are CALC, which loads a pseudo-register with a value or memory location to be used in math calculations; PLUS, which adds a value to the one used in CALC and STORE, which stores the result of the calculations in a specified memory location (word). (lines 240 to 290)

Line 580 tests if XC is greater than 240. If so, the program jumps to AGAIN. See if you can figure out what the program does. It uses the hybrid mode 10/8

Next time, we'll have some graphics macros that allow to translate and rotate objects.

TOUCH TABLET APPLICATIONS

by Ian Champ

Those of you who have purchased the Atari Touch Tablet or another touch tablet will already know how useful a tablet is for putting graphics onto the screen, but do you realise that it is also a particularly useful instrument to use in your own programs?

Before you can use the tablet in your own programs, you will need to know how it sends information to the computer. Type in listing 1 and try pressing the tablet in various spots (note: these programs are for the Atari Touch Tablet, but I would not be at all suprised if they work in much the same manner with the Koala Pad). You should see two columns of numbers. As you move your finger from left to right on the tablet the number in the left column increases. As your move your finger from bottom to top the number in the right hand column increases.

```
5 ? "↖"
10 IF PEEK(624)=228 AND PEEK(625)=228
THEN 10
20 ? PEEK(624),,PEEK(625)
30 GOTO 10
```

Thus you can see from studying listing 1 that the tablet is continually sending numbers to memory locations 624 and 625 to indicate the vertical and horizontal points at which it is being pressed. If it is not being pressed at all, it will send the number 228 to both memory locations (hence the use of line 10 in listing to stop any numbers appearing on the screen if the tablet is not pressed).

What about the two buttons on the tablet, and the other on the stylus? Type **NEW** then type:-

```
10 PRINT STICK(0):GOTO 10
```

and RUN this program. You should get 15 if none of the buttons are pressed, and various other numbers when one or a combination of the buttons is pressed.

Using listing one, numbers between 0 and 228 were obtained, however it is almost impossible to be accurate enough to be able to point to exactly the same spot twice,so I decided to split the tablet into a series of boxes. I ruled a piece of paper which was exactly the same size as the surface of the tablet into 25 boxes across, and 18 boxes up (these are arbitrary numbers, and infact a piece of graph paper would do just as well).

I put this under the plastic on the tablet and using listing one, I established the smallest numbers that I could get (using a sharp point this time rather than pointing with a finger). The numbers that I got were 4 (across value) and 1 (up value), but you will probably get different numbers with your tablet. I then went to the box on the top right corner of my sheet, and placed the point on the lower left corner of this box, and noted these values. I got 212 (across) and 225 (up) but again, your values might be different. These values are used in lines 20 and 25 of listing 2, ie, 212/25 boxes=8.48 and 225/18 boxes=12.5. The net result of this is that listing 2 can print on the screen the number of the box being pointed to. This is useful because while with listing 1 it is extremely difficult to point to any particular point, using listing 2, it is quite easy to point to a particular box at will.

```
10 ? "↖"
15 IF PEEK(624)=228 THEN 15
20 ? INT((PEEK(624)-4)/8.5);" ACROSS",
30 ? INT((PEEK(625)-1)/12.5);"UP      "
40 GOTO 15
```

By expanding this program I have for example put a sheet on the tablet which shows large calculator buttons, and have written a short program which prints the number pressed on the screen, and with a little more work could become a simulated calculator. It would also be possible to draw a piano type keyboard and put this onto the tablet, and to write a program so that one could play a tune by pointing to simulated keys! The possibilities are only as limited as your imagination, the tablet might be used as the console of a rocket ship or as a large key input device for children or those with vision or coordination difficulties.

In visualising the tablet as a game controller, it is interesting to note that memory locations 624 and 625 are also used by the paddle controllers. If you have a game that uses paddles, try playing it by pointing to the tablet, but be warned, you may decide to throw away your paddles after having done so! If there are quite a few people out there who have the Touch Tablet, and if they are interested enough to let me know that they exist (via ACE), then I will design some applications and underlays that could be used with the tablet. Perhaps someone else could do likewise for the Koala Pad.

Because it is possible to translate the pictures made by tablets, lightpens and joysticks (see my other article and

➡

the conversion program in Antic Vol. 3, No. 5), I would be very interested in trading pictures a diskful at a time with others. I would suggest that the best way to do this is through the clubs software exchange, and that the club should encourage this to happen. Perhaps the club could soon be staging art exhibitions for the public!

Finally then, try listings 3 and 4 and see if you can build on them to turn the tablet into a unique musical instrument.

```
5 REM **LISTING 3**
10 IF PEEK(624)<228 AND PEEK(625)<228
THEN SOUND 0,PEEK(624),10,PEEK(625)/15
:GOTO 10
20 SOUND 0,0,0,0:GOTO 10
```

```
5 REM **LISTING 4**
10 IF PEEK(624)<228 AND PEEK(625)<228
THEN SOUND 0,PEEK(624),10,10:SOUND 1,P
EEK(625),10,10:GOTO 10
20 SOUND 0,0,0,0:SOUND 1,0,0,0:GOTO 10
```

BOOK REVIEW

THE ATARI 800XL by THOMAS BLACKADDER

Reviewed by Dennis Sampson

I found this book extremely easy to read and most importantly, easy to use with my Atari 800XL. The book skips across the wide spectrum of 'uses' without 'bogging down' at any particular stage.

For the beginner (like me), I think this book is a very worthwhile addition to any computer library. Capital letters are used to emphasise key words, while italics made for easy references and negated the use of 'highlighter' pens.

The only criticism I can level at this book is the use of two chapters (9 and 10) devoted to storage: cassette and disk, since sufficient and comprehensive documentation is provided with the purchase of these hardware.

LABEL PRINTER

by K. J. Bricknell

LABEL PRINTER is a program to format and print text on fanfold sticky labels. Although written for the ATARI/EPSON combination and the ubiquitous 23mm by 90mm label size, the program could easily be modified for other dot matrix printers and label sizes.

Firstly, the code related to the rainbow-graphics title screen does nothing really useful. If you don't need this bit of decoration, simply delete line 20 and all lines after line 1880.

Run the program and the first thing you will be asked is whether you need instructions. After paging through the instructions (or after you answer 'no' to the request), you will be asked for the number of lines you require on the label. Any number of lines up to five may be input. (During printout, the top and bottom margins and the line spacing will be set according to the number of lines you input.)

You will then be led through the formatting of each line. For each label line you will be asked for the lettering size required, the actual text for that line, whether you wish the text automatically centred and whether you require normal, emphasized or double-strike print. An input mask shows you the maximum characters for each line. (This will, of course, vary according to the lettering size you select.)

If you make a mistake while typing the text line, step back to the required position with the DELETE BACK S key. Other input mistakes can be corrected by using the ESCape key. In fact, you can backtrack right back to the beginning of the format routine with successive presses of the ESCape key if you so desire. (The ESCape key may also be used in this way to correct input errors in other parts of the program.)

After you have formatted each label line, you will be asked if there is any particular line that you wish to have change automatically during printout. (This option is useful for, say, children's schoolbook labels where the name remains the same but the subject title changes from label to label.)

If you choose this option, you will be asked to input the text for the various versions of this line. The number of labels printed during printout will be determined by the number of versions of this line that you input. (If 30 labels are insufficient for your perceived needs, TEXTC$ (line 790) may be redimensioned.) If you do not choose this option, you will be asked to input the number of labels to be printed. You will then be led through the printout phase after which, if you wish, you will be returned to the formatting routine for the next label.

➸

```
1 REM ########################################
2 REM #          LABEL PRINTER             #
3 REM #      by K. J. Bricknell            #
4 REM # Published by Atari Computer #
5 REM #     Enthusiasts (N.S.W.)           #
6 REM #              June 1985             #
7 REM ########################################
30 OPEN #2,4,0,"K:"
40 GRAPHICS 0:POKE 82,0:SETCOLOR 1,0,1
4:SETCOLOR 2,0,0:POKE 752,1
50 POSITION 5,12:? "NEED INSTUCTIONS?
( Y =YES)":GET #2,A:IF A=89 THEN 1710
60 REM Initialization
70 DIM LSON$(10),LSOFF$(10),PTON$(10),
PTOFF$(10),MASK$(55),CBACK$(55),TEXT1$
(54),TEXT2$(54),TEXT$(270),L$(40)
80 DIM A$(2):PR=0:CN=0:CLN=0:AC=0:CHEC
K=0
90 L$(1)="—":L$(40)=L$(1):L$(2)=L$(1)
100 MASK$="█":MASK$(55)=MASK$(1):MASK$
(2)=MASK$(1)
110 CBACK$="←":CBACK$(55)=CBACK$(1):CB
ACK$(2)=CBACK$(1)
120 TEXT$=" ":TEXT$(270)=TEXT$(1):TEXT
$(2)=TEXT$(1)
130 ? "K";:SETCOLOR 2,2,4:SETCOLOR 4,2
,4:SETCOLOR 1,2,14:GOSUB 1600
140 REM Input number of lines
150 POSITION 0,11:? "How many lines th
is label? (Max 5)     ";
160 GET #2,A:IF A<49 OR A>53 THEN ? CH
R$(253);:GOTO 160
170 ? CHR$(A+128):NL=A-48
180 REM Main input loop
190 FOR L=1 TO NL:GOSUB 1610
200 SETCOLOR 2,6,4:SETCOLOR 4,6,4:SETC
OLOR 1,6,14
210 ? "K OK. Let's format line ";CHR$(
L+176);"                    ";:FOR D=1 TO
50:NEXT D:GOSUB 1600
220 REM Input lettering size
230 POSITION 0,2:? "Which lettering si
ze:":? :? "1NORMAL 2SMALL 3LARGE 4MEDI
UM          ";
240 GET #2,A
250 IF A=27 AND L=1 THEN GOSUB 1660:GO
TO 130
260 IF A=27 AND L>1 THEN L=L-1:GOSUB 1
660:GOTO 200
270 IF A<49 OR A>52 THEN ? CHR$(253);:
GOTO 240
280 ? CHR$(A+128);:LS=A-48
290 IF LS=1 THEN LSON$(L*2-1,L*2)="♥♥"
:LSOFF$(L*2-1,L*2)="♥♥":MASKLEN=31:GOT
0 330
300 IF LS=2 THEN LSON$(L*2-1,L*2)="▪♥"
:LSOFF$(L*2-1,L*2)="—♥":MASKLEN=54:GOT
0 330
310 IF LS=3 THEN LSON$(L*2-1,L*2)="_♥"
:LSOFF$(L*2-1,L*2)="●♥":MASKLEN=16:GOT
0 330
320 IF LS=4 THEN LSON$(L*2-1,L*2)="▪_"
:LSOFF$(L*2-1,L*2)="●—":MASKLEN=27
330 GOSUB 1600
340 REM Input text for line
350 POSITION 0,6:? L$:? "Text for line
 ";L;":"
360 POSITION 0,10:? MASK$(1,MASKLEN);C
BACK$(1,MASKLEN);:IF MASKLEN=54 THEN ?
 "↑";
370 FOR X=1 TO MASKLEN
380 GET #2,A:IF A=27 AND X=1 THEN GOSU
B 1660:GOTO 200
390 IF A=126 AND X=1 THEN GOTO 360
400 IF A=126 AND X=41 THEN X=X-1:TEXT1
$(X)="":? "←↑█←↑";:GOTO 380
410 IF A=126 THEN X=X-1:TEXT1$(X)="":?
 "←█←";:GOTO 380
420 IF A=155 AND X=1 THEN TEXT1$=" ":G
OTO 480
430 IF A=155 THEN 480
440 IF A=27 THEN TEXT1$="":GOTO 360
450 TEXT1$(X)=CHR$(A):? CHR$(A);
460 NEXT X
480 GOSUB 1600
490 REM Input centering choice
500 POSITION 0,13:? L$:? "Automatic ce
ntering (Y=Yes N=No)        ";
510 GET #2,A
520 IF A=27 THEN FOR D=13 TO 16:POSITI
ON 0,D:? "
              ";:NEXT D:GOTO 350
530 IF A<>89 AND A<>78 THEN ? CHR$(253
);:GOTO 510
540 ? CHR$(A+128);:IF A<>89 THEN HSL=1
:TEXT2$(HSL)=TEXT1$:GOTO 560
550 GOSUB 1560
560 TEXT$(L*54-53,L*54)=TEXT2$
570 GOSUB 1600
580 REM Input print type
590 POSITION 0,17:? L$:? "Which print
type:":? :? "1NORMAL 2EMPHASIZED 3DOUB
LE STRIKE     ";
600 GET #2,A
610 IF A=27 THEN GOSUB 1660:GOTO 200
620 IF A<49 OR A>51 THEN ? CHR$(253);:
GOTO 600
630 ? CHR$(A+128);:FOR D=0 TO 70:NEXT
D
640 PT=A-48:? "K"
650 IF PT=1 THEN PTON$(L*2-1,L*2)="♥♥"
:PTOFF$(L*2-1,L*2)="♥♥":GOTO 700
660 IF PT=2 AND LSON$(L*2-1,L*2)="▪♥"
THEN PTON$(L*2-1,L*2)="♥♥":PTOFF$(L*2-
1,L*2)="♥♥":GOTO 700
670 IF PT=2 AND LSON$(L*2-1,L*2)="▪_"
THEN PTON$(L*2-1,L*2)="♥♥":PTOFF$(L*2-
1,L*2)="♥♥":GOTO 700
680 IF PT=2 THEN PTON$(L*2-1,L*2)="ξE"
:PTOFF$(L*2-1,L*2)="ξF":GOTO 700
690 IF PT=3 THEN PTON$(L*2-1,L*2)="ξG"
:PTOFF$(L*2-1,L*2)="ξH"
700 IF CHECK=1 THEN CHECK=0:GOTO 720
710 NEXT L
720 GOSUB 1610:SETCOLOR 2,10,4:SETCOLO
R 4,10,4:SETCOLOR 1,10,14
730 REM Changing line module
740 ? "KIs there a particular line tha
t you wishto automatically change from
 label to   label during printing ";
750 ? " (Y=Yes)          ";
760 GET #2,A:IF A=27 THEN L=NL:CHECK=1
:GOSUB 1660:GOTO 200
770 IF A<>89 AND A<>78 THEN ? CHR$(253
);:GOTO 760
780 IF A<>89 THEN 1110
790 PR=1:CN=2:DIM TEXTC$(1620):GOSUB 1
610:SETCOLOR 2,4,2:SETCOLOR 4,4,2
800 ? "KOK. Which line is the one whic
h has to  change from label to label?
          ";
810 GET #2,A:IF A<49 OR A>NL+48 THEN ?
 CHR$(253);:GOTO 810
820 ? CHR$(A+128):CLN=A-48:TEXTC$=" ":
TEXTC$(1620)=" ":TEXTC$(2)=TEXTC$(1)
830 TEXTC$(1,54)=TEXT$(CLN*54-53,CLN*5
4)
840 IF LSON$(CLN*2-1,CLN*2)="♥♥" THEN
MASKLEN=31
850 IF LSON$(CLN*2-1,CLN*2)="▪♥" THEN
MASKLEN=54
860 IF LSON$(CLN*2-1,CLN*2)="_♥" THEN
MASKLEN=16
870 IF LSON$(CLN*2-1,CLN*2)="▪_" THEN
MASKLEN=27
880 GOSUB 1610:? "KOK. We already have
 the text for the    first line ";CLN;
". Now type the text"
890 ? "for each successive line ";CLN;
" followed by  RETURN":? :? "Hit RETUR
N at input prompt to terminate"
900 ? :? "Incidentally, do you want th
ese inputs  automatically centred (Y=Y
es)          "
910 GET #2,A:IF A<>89 AND A<>78 THEN ?
 CHR$(253);:GOTO 910
920 IF A<>89 THEN 940
930 ? :? "It shall be done":AC=1
940 GOSUB 1610:? :? "OK. Start your in
puts":? :?
950 ? MASK$(1,MASKLEN);CBACK$(1,MASKLE
N);:IF MASKLEN=54 THEN ? "↑";
960 FOR X=1 TO MASKLEN
970 GET #2,A:IF A=155 AND X=1 THEN 110
0
980 IF A=126 AND X=1 THEN GOTO 950
990 IF A=126 AND X=41 THEN X=X-1:? "←↑
█←↑";:GOTO 970
1000 IF A=126 THEN X=X-1:? "←█←";:GOTO
 970
1010 IF A=27 AND MASKLEN=54 AND X<40 T
HEN ? "↑":GOTO 950
1020 IF A=27 AND MASKLEN=54 AND X>40 T
HEN ? "↑↑":GOTO 950
1030 IF A=27 AND MASKLEN<>54 THEN ? "↑
":GOTO 950
1040 IF A=155 THEN 1070
1050 TEXT1$(X)=CHR$(A):? CHR$(A);
1060 NEXT X
1070 IF AC<>1 THEN TEXT2$=TEXT1$
1080 IF AC=1 THEN GOSUB 1560
1090 TEXTC$(CN*54-53,CN*54)=TEXT2$:CN=
CN+1:? :? :GOTO 950
1100 CN=CN-1:? "K":GOTO 1220
1110 GOSUB 1600:SETCOLOR 2,3,4:SETCOLO
R 4,3,4
1120 REM Input number of copies
1130 ? "KHow many copies this label
          █←←↑";
1140 FOR X=1 TO 2:GET #2,A
1150 IF A<48 OR A>57 AND A<>155 THEN ?
 CHR$(253);:GOTO 1130
1160 IF A=27 THEN 1130
1170 IF A=155 AND X=1 THEN 1130
1180 IF A=155 AND X=2 THEN 1210
1190 A$(X,X)=CHR$(A):? CHR$(A+128);
1200 NEXT X
1210 B=VAL(A$):CN=B
1220 GOSUB 1600
1230 REM Print routine
1240 SETCOLOR 2,3,4:SETCOLOR 4,3,4:? :
? "KOK. Set the label fanfold in the p
rinterso that :"
1250 ? :? "1.  The top of the carbon r
ibbon is     aligned with the space be
tween the      labels, and"
1260 ? :? "2.  The left edge of the la
bel is       aligned with the second l
ine on the     scale"
1270 ? :? :? "Ensure printer is on-lin
e and hit RETURNwhen ready to print":?

1280 GET #2,A:IF A=27 THEN 1130
1290 IF A<>155 THEN ? CHR$(253):GOTO 1
280
```

➔

```
1300 GOSUB 1600:? "KPrinting...":?
1310 TRAP 1630:OPEN #1,8,0,"P:":? #1;C
HR$(27);"@";CHR$(27);"C";CHR$(0);CHR$(
1);CHR$(27);"8";
1320 IF NL=3 THEN ? #1;CHR$(27);"A";CH
R$(18);
1330 IF NL=2 THEN ? #1;CHR$(27);"A";CH
R$(24);
1340 IF NL=5 THEN ? #1;CHR$(27);"A";CH
R$(10);
1350 FOR LL=1 TO CN
1360 IF NL=5 THEN ? #1;CHR$(27);"J";CH
R$(18);
1370 IF NL=4 OR NL=3 THEN ? #1;CHR$(27
);"J";CHR$(18);
1380 IF NL=2 THEN ? #1;CHR$(27);"J";CH
R$(36);
1390 IF NL=1 THEN ? #1;CHR$(27);"J";CH
R$(72);
1400 FOR L=1 TO NL
1410 ? #1;PTON$(L*2-1,L*2);LSON$(L*2-1
,L*2);
1420 IF PR=1 AND L=CLN AND LSON$(L*2-1
,L*2)="_♥" THEN ? #1;TEXTC$(LL*54-53,L
L*54-38);:GOTO 1460
1430 IF PR=1 AND L=CLN AND LSON$(L*2-1
,L*2)<>"_♥" THEN ? #1;TEXTC$(LL*54-53,
LL*54);:GOTO 1460
1440 IF LSON$(L*2-1,L*2)="_♥" THEN ? #
1;TEXT$(L*54-53,L*54-38);:GOTO 1460
1450 IF LSON$(L*2-1,L*2)<>"_♥" THEN ?
#1;TEXT$(L*54-53,L*54);
1460 ? #1;LSOFF$(L*2-1,L*2);PTOFF$(L*2
-1,L*2)
1470 NEXT L:? #1;CHR$(12);:? "Label nu
mber ";LL;" done"
1480 NEXT LL:? #1;CHR$(7);:CLOSE #1
1490 ? :? "All Done!":FOR D=1 TO 200:N
EXT D:GOSUB 1600
1500 REM Any more?
1510 ? :? "Any more (Y=Yes)
          ";
1520 GET #2,A:IF A<>89 AND A<>78 THEN
? CHR$(253);:GOTO 1520
1530 IF A=89 THEN CLR :? "K":GOTO 70
1540 GRAPHICS 0:POKE 82,2:END
1550 REM Text centering subroutine
1560 B=LEN(TEXT1$):IF B>MASKLEN-2 THEN
 TEXT2$=TEXT1$:RETURN
1570 HSL=(MASKLEN-B)/2:TEXT2$=" ":TEXT
2$(HSL)=TEXT2$(1):TEXT2$(2)=TEXT2$(1)
1580 TEXT2$(HSL+1)=TEXT1$:RETURN
1590 REM Sound subroutines
1600 FOR S=1 TO 15:SOUND 1,RND(0)*10+5
,10,14:NEXT S:SOUND 1,0,0,0:RETURN
1610 FOR S=1 TO 15:SOUND 1,RND(0)*10+3
0,10,14:NEXT S:SOUND 1,0,0,0:RETURN
1620 REM Error handler for printer not
 on-line
1630 CLOSE #1:? CHR$(253);"KPrinter no
t on-line. Place on-line and  hit RETU
RN":GET #2,A:IF A<>155 THEN 1630
1640 GOTO 1310
1650 REM Start again subroutine
1660 SETCOLOR 2,3,4:SETCOLOR 4,3,4:? "
K":POSITION 7,11
1670 IF L=1 THEN POSITION 9,11:? "OK.
START LABEL AGAIN"
1680 IF L>1 THEN POSITION 9,11:? "OK.
START LINE ";L;" AGAIN"
1690 GOSUB 1610:FOR D=0 TO 70:NEXT D:R
ETURN
1700 REM Instuctions
1710 ? "KThis program allows you to fo
rmat print-outs on fanfold label paper
 using an    EPSON MX80 printer."
1720 ? :? "You may have any number of
lines up to  a maximum of five."
1730 ? :? "You may have the text for a
ny one line  in normal, small, large o
r medium       lettering."
1740 ? :? "You may have the print for
any one line in normal, emphasized or
double strike."
1750 ? "(Note that there is no emphasi
zed print with small and medium letter
ing. If you"
1760 ? "input emphasized print with sm
all and   medium lettering, your input
 will be    disregarded.)"
1770 ? :? "You may have the text on a
line automat-ically centered if you wi
sh."
1780 POSITION 0,23:? "Hit any key to c
ontinue";:GET #2,A:GOTO 1790
1790 ? "KAn input mask will show you h
ow many    characters will fit on one
line for any lettering size."
1800 ? :? "If you make a mistake while
 inputting   responses, hit ESC to sta
rt again."
1810 ? :? "An option is available wher
eby one line changes its text from one
 label to the  next.":?
1820 ? "If you choose this option, the
 number   of labels printed will be au
tomatically determined by the";
1830 ? " number of extra       version
s of this line that you input."
1840 ? :? "If you do not choose this o
ption, you   will be asked to input th
e specific num-ber of copies you ";
1850 ? "wish to print."
1860 POSITION 0,23:? "Hit any key to c
ontinue";:GET #2,A:GOTO 1870
1870 ? "KOK. From now on just answer t
he quest-  ions and provide the inputs
  requested."
1880 POSITION 0,23:? "Hit any key to c
ontinue";:GET #2,A:GOTO 70
```

OOPS ! ! !

Well it seems a familiar operating system bug has managed to camouflage itself within the last issue, INSIDE INFO No.18.

In BOOT DISK UTILITY, the line:- "...XL users will need to change the JSR in line 2110..."
Should be:- "...Atari 400 and 800 owners will need to change the JSR in line 2110..."

And in the listing, line 20100 should be:-

```
20100 DATA 248,24,165,4,105,128,133,4,165,5,105,0,133,5,206,65,2,240,18,238,10,3,208,3
```

* * * * *

MORE PROGRAMS...

Believe it or not, some members take the effort to submit articles for INSIDE INFO. Some of the program submissions are extremely large and would take up a substantial portion of INSIDE INFO. Yet, these programs are of exceptional worth that I felt that you, as readers should know about them. These programs will become available from the Bulletin Board or Software Exchange. My apologies to the authors concerned for being unable to print the whole article.

COMOKILL by Glynn Gough is a short demo that'll blow your Commodore owner next door away. You just can't help but to smile in satisfaction. Any Atari loyalist should have this by his side. Definitely the perfect shop window display.

CHECKBOOK RECONCILER by Andrew Powell does what its name implies. By entering your cheque transactions the program creates a reconciled cheque balance which may be examined against your latest Bank Statement.

*****Atari Computer Enthusiasts (New South Wales)*****

INFORMATION JUNE 1985

Atari Computer Enthusiasts (N.S.W.) is an independent non-profit computer users' group loosely affiliated with Eugene A.C.E in the USA. and with no connection to ATARI Inc. Membership fees are currently $30 for the first year and $15 (Renewal) thereafter. Membership permits free access to the group's Bulletin Board service and a 6 issue subscription to INSIDE INFO. Unless noted as being copyright, permission is given to reprint articles from INSIDE INFO, making acknowledgement to the source. Opinions expressed are those of the author and do not necessarily represent the views of ACE (NSW).

COMMITTEE MEMBERS	HOME PHONE NO.	COMMITTEE MEMBERS	HOME PHONE NO.
Barry Williams (President)	(02) 452 2229	Tony Reeve (Vice President)	
Alex Kwok (Secretary/Treasurer)	(02) 661 3377	Jamie Athas (Meetings Organiser)	(02) 349 7365
Ken Shiu (Editor)	(02) 534 2120	Roberto Romano (Program Editor)	(02) 327 4898
Jeff Maddock (Bulletin Board)	(02) 568 2990	Peter Dickeson (Librarian)	(02) 80 2505

PAYMENTS, CORESPONDENCE AND NEWSLETTER EXCHANGE

All payments may be made at any meeting or by mail. All cheques and Money Orders should be made payable to Atari Computer Enthusiasts (N.S.W.). All correspondence and Newsletter Exchange should be forwarded to:-

A.C.E. (N.S.W.) GPO BOX 4514, SYNDEY N.S.W. 2001

BULLETIN BOARD - Phone (02) 560 9846

Operating hours - 6PM to 9AM Monday to Friday. Full 24 hours on both Saturday and Sunday.
Access requires a personal password from SYSOP upon membership. 'AMODEM' terminal software or derivatives is recommended. PUBLIC DOMAIN SOFTWARE ONLY!!

MEMBERS' DISCOUNTS

COMPUTER-1 [5% Discount]:- 200 Alison Road , Randwick. Phone:- (02)399 8865
HOTLINE BOOKS AND SOFTWARE [10% Discount]:- 693 George Street, Haymarket, Sydney. Phone:- (02)212 6077
COMPUTERWAVE [5% Discount -Cash Only]:- 325 George Street, Sydney (near Wynyard). Phone:- (02) 29 1631
THE COMPUTER SPOT [5% Credit, 10% Cash]:- Shop C4, MLC Centre, Martin Place. Phone (02)235 2971
Eastgate Centre, Bondi Junction. Phone:- (02)387 5208. Shop 21A, Greenway Arcade, Parramatta. Phone:- (02)498 5631.
COVER CARE [25% off computer dust covers]:- P.O.Box 719, Chatswood, N.S.W. 2067. Phone:- (02)498 5631

ARTICLE/PROGRAM SUBMISSIONS

Articles for INSIDE INFO should reach the Editor as early as possible, and at least ONE MONTH ahead of the release date. Disk submissions should use single density DOS 2 format. Submissions will entitle you to one Software Exchange disk at half price.

MEETING DETAILS

Meetings are held at 6PM in the 2nd Floor of the Y.W.C.A, corner of Wentworth Avenue and Liverpool Street, Darlinghurst 2010.

UPCOMING MEETINGS:- TUESDAY JUNE 11
MONDAY JULY 1
MONDAY AUGUST 12

Atari Computer Enthusiasts (N.S.W.)

MEMBERSHIP FORM

NAME: ______________________________

ADDRESS: ______________________________

TOWN/SUBURB: ______________________________

STATE:- ____________________ *POSTCODE:* ____________

PHONES:- home ( ) - ______ *work* ( ) - ______

SIGNATURE: ____________________ *AGE:* ______ *DATE:* / /

Enclosed is []Cash []Cheque []Money Order for:-
[]$20 joining membership []$15 membership renewal []Notice of change of address

OFFICE USE ONLY:
Date Processed: / / *Receipt #:* ________ *Last Issue:* ________

FOR SALE

DISK SOFTWARE:- Bruce Lee, Summer Games, Starfire/ Fire One, Jumpman, O'Riley's Mine, Stellar Shuttle, Murder on the Zinderneuf, Adventure Master (CBS), Choplifter, Skyblazer, Lode Runner, Snooper Troops (Granite Point Ghost).
CARTRIDGES:- Pengo, Defender, River Raid, Star Raiders, Asteroids, Kickback, Learning with Leeper.
CASSETTE:- Zaxxon
ALL ARE ORIGINAL AND IN NEW CONDITION WITH DOCUMENTATION, EACH FOR APPROX. $20. CONTACT:- LINDSAY HERON, NEWCASTLE, Ph. (049) 58 1049

* * * * *

CARTRIDGES:- AtariWriter $45 each
DISK SOFTWARE:- Synfile (new condition in box), $75 each
CONTACT:- LAWRENCE LAM, Ph. (02) 643 2742

* * * * *

INSIDE INFO No.1 and No.3 photocopies no longer needed. 50 cents (plus postage). PHONE:- (02) 623 7393

* * * * *

PRINTER SYSTEM:- Atari 822 printer. 4 rolls of paper, software to screen dump and fancy characters. NO interface required. Whole system, $80. CONTACT:- Ken Scalley, Winston Hills, Ph.(02) 624 4561

WANTED

ZORK I: CONTACT:- Glynn Gough, 176 Bargara Road, Bundaberg, Qld. 4670. Ph. (071) 71 5171

* * * * *

ANALOG, ISSUE 16: CONTACT:- J.Nolan Ph. (02) 626 8998

* * * * *

DOES anyone have a program to run the graphics for MR T from the 'A-Team'. My school kids are driving me crazy to see him on the screen. CONTACT:- Paul Sweeney. 49 Ingleburn Road, Leppington 2171. Ph. (02) 606 5036.

* * * * *

PROGRESSIVE COMPUTING PTY LTD
P.O. BOX 243, ENFIELD, NSW 2136 TELEPHONE (02) 76-5615

BASIC XL RRP $129.00
This is the BASIC Atari should have used. Long programs can run 2-4 times faster. Special commands for Player/ Missiles, strings, trace, auto line numbering, DOS functions & more.

ACTION! RRP $149.00
High level, structured, compiled language for Atari. For those who have found BASIC too slow or assembler too difficult. Fast, powerful and user friendly.

MAC/65 RRP $149.00
A fast, powerful macro assembler.
Includes Editor, Macro assembler and debugger.

WRITERS TOOL RRP $149.00
Designed for writers who want to write, not spend hours learning to use a word processor. Use all your printer's capabilities when printing. Includes spelling checker.

TOOL KITS & DOS XL RRP $69.95
Special library of macros for MAC/65.
BASIC XL kit-extended commands + demo games
ACTION! kit-library of subroutines + demo games

R-VERTER RRP $89.00
Connect most modems (those with DB25 socket) to the ATARI without the 850 interface. Connects to standard I/O port. Includes RS232 handlers, smart terminal software with XMODEM protocol, even 80 column display.

MPP Product
Print Buffer RRP $329
For all computers with centronics output, 64K, self test, clear button, copy repeat button.

Printer inerfaces - connects to standard I/O port, compatible with all software. MICROPRINT $119, MPP-1150 (with I/O port) $149

ANTIC single issue $8.50
1 year subscription $100.00
ANALOG - call for availability

DEALER ENQUIRIES WELCOME

Call for your nearest dealer or send cheque, money order or Bankcard number (with name as appears on card and validity dates).
P&P - $4.00 1st item, $1.00 ea. item thereafter.
Prices subject to change without notice.